

## Google Design Sprint como um Recurso Educacional: uma Pesquisa Exploratória

Vinícius Gomes Ferreira e Edna Dias Canedo

Departamento de Ciência da Computação, Mestrado em Computação Aplicada (PPCA), Universidade de Brasília (UnB), P.O. Box 4466

Brasília, Brasil

vinicius.gomes@ifg.edu.br, ednacanedo@unb.br

### ABSTRACT

As technology undergraduates target the market and they have realized the need for agile and innovative processes, the insertion of User Centered Design techniques is a great gain to their repertoire. This article sought to verify how Design Sprint behaves into reduced number of classes when it must generate a functional prototype. For this, Design Sprint was applied as a case study at the Federal Institute of Goiás. In the questionnaire applied, 91% of students rated Design Sprint as productive in terms of learning and 100% of students agreed that Design Sprint can creatively solve problems and they expressed the desire to participate in new applications of it. After the case study was applied, it is inferred that Design Sprint generate interesting insights that can be useful in a project-based learning context and students judge their own learning and participation to be positive in Design Sprint based classes.

### Author Keywords

Design Sprint; Active Learning; Software Engineering; Education; prototyping.

### ACM Classification Keywords

• Applied computing ~ Interactive learning environments • Software and its engineering ~ Software prototyping

### INTRODUCTION

Para que as empresas possam diferenciar seus produtos e oferecer serviços que realmente agreguem valor ao cliente torna-se cada vez mais necessário incorporar dois componentes básicos ao seu processo de desenvolvimento: agilidade e inovação. Entretanto, apenas ter boas ideias apenas não é o suficiente. Mesmo as melhores ideias enfrentam um caminho bastante incerto no percurso até o

Paste the appropriate copyright/license statement here. ACM now supports three different publication options:

- ACM copyright: ACM holds the copyright on the work. This is the historical approach.
- License: The author(s) retain copyright, but ACM receives an exclusive publication license.
- Open Access: The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

This text field is large enough to hold the appropriate release statement assuming it is single-spaced in Times New Roman 8-point font. Please do not change or modify the size of this text box.

Each submission will be assigned a DOI string to be included here.

sucesso na vida real [1]. Mesmo as *startups*, empresas com foco em inovação, fracassam e grande parte dos seus produtos não fazem o sucesso esperado quando lançados no mercado [2]. E, desta forma, expectativa da indústria recai nas universidades, visto que a qualidade do exercício da função de um bom engenheiro de *software* depende da educação que este recebeu [3].

Existem muitos estudos que indicam que os alunos do século XXI necessitam competências que ultrapassem o nível técnico [4], [6]. Algumas dessas competências são encontradas nos modelos de pensamento projetual (*Design Thinking*) [7],[9].

O *Design Thinking* (DT) pode ser entendido como um conjunto de ferramentas e técnicas usados por um time multidisciplinar que, aplicadas à fase de design, visa estimular e promover a inovação e transformação das organizações [10], [11]. Depois do DT, surgiram outras ferramentas metodológicas que não apenas identificam ideias, mas também as desenvolvem até que possam ser testadas com os usuários finais. Foi o caso da *Lean Startup* [2] e com o mesmo objetivo em mente, mas fazendo um processo similar em um prazo de cinco dias, a *Google Design Sprint*, que foi desenvolvida nos laboratórios da Google Ventures [1].

Não apenas habilidades de desenvolvimento são suficientes para as futuras equipes ágeis, mas também a integração do processo de *Design Centrado no Usuário*. Por isso, habilidades inovadoras em experiência de usuário (UX) usadas nas metodologias mencionadas acima também são desejáveis aos alunos de graduação em áreas de computação [12], [13]. No trabalho apresentado por Hardy, Myers e Sankupellay [7], os autores perceberam esta necessidade e desenvolveram uma maneira de inserir o DT em uma universidade na Austrália. Com base na mesma premissa, o DT também está sendo integrado na grade curricular de cursos superiores dos Estados Unidos. [8]. A internalização da necessidade de uma proximidade com o cliente, um aspecto verificável no uso da *Google Design Sprint* [14], é um ganho potencial muito valioso aos estudantes de uma graduação em tecnologia.

No entanto, a realidade local do Instituto Federal de Educação, Ciência e Tecnologia - Campus Formosa

permanece distante. Os alunos da instituição possuem pouco conhecimento sobre *Design Thinking*, *Lean Startup*, *Google Design Sprint* e todas estas novas técnicas empreendedoras de UX.

Motivados por essas questões, fizemos um estudo exploratório para que, a partir dos dados obtidos, seja possível propor práticas de ensino de técnicas de UX no contexto educacional do Instituto Federal de Goiás. Neste experimento, optamos pelo uso da *Google Design Sprint*, que é o “**Grande sucesso** de estratégia de negócios, inovação, ciência comportamental e muito mais - incluídos em um processo passo a passo que qualquer equipe pode usar” [1]. Nós fizemos adaptações na técnica a fim de que viesse a ter uma abordagem minimalista, para ser aplicada dentro de um encontro semanal com os alunos da disciplina de Engenharia de Requisitos do curso de Tecnologia em Análise e Desenvolvimento de Sistemas (TADS) do Instituto Federal - Campus Formosa-GO. A escolha dessa turma se deu por causa da afinidade do conteúdo das aulas dessa disciplina com a abordagem da *Google Design Sprint*.

No relato apresentado neste trabalho, utilizamos alguns termos que gostaríamos de convencionar para melhor entendimento. Com a palavra curso, queremos nos referir ao curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia de Goiás - Campus Formosa, ao qual também nos referimos simplesmente como TADS. Ao nos referirmos ao conjunto de duas aulas semanais da disciplina de Engenharia de Requisitos, para o qual a *Google Design Sprint* foi adaptada, usamos o termo encontro semanal. À Coordenação de Integração Escola - Empresa responsável pelos processos de estágio chamamos Coordenação de Integração Escola – Empresa (COSIE-E) e ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás chamamos de IFG e, às vezes, também nos referimos ao termo *sprint* como sendo a *Google Design Sprint*.

## FUNDAMENTAÇÃO TEÓRICA

### *Google Design Sprint*

Um dos problemas dos *workshops* e outras técnicas de discussão em grupo, é que normalmente as ideias que seriam bem sucedidas não eram aquelas geradas durante uma barulhenta sessão de *brainstorming* [1], mas em momentos de introspecção dos indivíduos. Conclui-se que as melhores ideias apareciam depois que a excitação dos *workshops* terminava, normalmente quando as pessoas continuavam a pensar enquanto estavam em seus escritórios, esperando por um café ou mesmo durante o banho [1]. O desafio, nesse caso, era trazer para um ambiente de negócios uma ferramenta que permitisse extrair dos indivíduos essas melhores ideias [1], [15].

Na sua atuação nos projetos da caixa prioritária do Gmail e de uma videoconferência que mais tarde viria a se tornar o Google Hangouts, ele notou que focar no trabalho individual, ter tempo para prototipar e trabalhar com uma data limite

rigorosa trazia resultados muito melhores que seus *workshops* tradicionais anteriormente usados. A essa maneira de se trabalhar, ele deu o nome de *Design Sprint* [1].

Em suma, o *Google Design Sprint* pode ser entendido como um processo bem delineado, composto de etapas e ferramentas para se obter soluções criativas e testá-las com os usuários finais. O primeiro estágio para sua execução é preparar o ambiente. Antes de iniciar uma *sprint*, é necessário certificar-se de que se dispõe do desafio correto, do time certo e de tempo e espaço físico para a condução da *sprint*. Quanto maior o desafio, melhor a *sprint*. Uma vez preparado o ambiente, seguem-se as etapas da *sprint* que ocupam dias inteiros durante uma semana útil, conforme detalhado na Figura 1.



Figura 1. Etapas da *Google Design Sprint*.

### Segunda-feira – Mapear/Entender

O propósito da segunda-feira é entender e mapear o objetivo da *sprint* e seus obstáculos do início ao fim. Para isso, são realizadas conversas com especialistas do negócio, de forma que todos os participantes tenham uma melhor compreensão do problema geral. Uma representação curta do processo é feita. Essa deve incluir todos os clientes e interessados-chave à esquerda e o objetivo final à direita, ao passo que, no meio seja desenvolvido um fluxo simplificado de cinco a quinze passos. Desse mapa desprende-se uma visão geral do caminho a ser percorrido para o objetivo final e será nele onde o time identificará pontos de melhoria no processo utilizando a técnica do *How Might We*. Essa é uma ferramenta em que cada integrante do time identifica problemas no fluxo e silenciosamente anota perguntas em cartões começando com as iniciais HMW, que significam "Como nós poderíamos", para as quais trarão soluções em um momento posterior. Tendo os cartões sido preenchidos pelo time, eles são colados em um quadro ou parede, organizados por grupos de ideias para que aqueles mais relevantes recebam votos. Feita a distribuição dos votos, a etapa seguinte se resume em passar os cartões mais votados para o mapa;

### Terça-feira – Criar/Esboçar

A terça-feira é o dia em que serão propostas soluções para os problemas identificados e priorizados na segunda-feira. Como fonte de inspiração para a equipe, o dia deve iniciar com uma revisão do que outras empresas ou provedores de serviços têm feito sobre determinado problema. Essa revisão, denominada pelo autor de *Lightning Demos* deve ser informal e cada exemplo deve ser apresentado em poucos minutos. À medida em que os exemplos são apresentados, as boas ideias são capturadas e anotadas em um grande quadro onde todos possam ver e acompanhar. Uma vez finalizada a

etapa de revisão, os participantes iniciam as propostas de solução por meio da criação de esboços. [1] sugerem uma técnica dividida em quatro etapas, na qual cada participante deverá executar, de forma individual, cada uma delas.

#### *Quarta-feira – Decidir*

Primeiramente, todas as soluções propostas no dia anterior são montadas em uma parede ou quadro. Então, cada participante deve analisar tais soluções em silêncio e usar marcadores adesivos para identificar as partes que julgar mais promissoras. Depois de feita esta análise individual, a equipe deve discutir cada esboço de solução em conjunto a fim de que tome notas das ideias mais relevantes. Ao final, cada participante poderá votar nas partes que considerar mais relevantes. Desta forma, os esboços que farão parte da solução final serão identificados e um *storyboard* será desenhado para planejar o protótipo a ser construído no dia seguinte.

#### *Quinta-feira - Prototipar*

O *storyboard* de como será o teste na sexta feira é criado na quarta-feira. Isso acontece para que durante a concepção do protótipo, dúvidas de como um esboço se encaixa no outro sejam um grande empecilho. Este protótipo a ser desenvolvido não precisa ser uma representação gráfica final, apenas um esboço, mas deve ser fiel o suficiente para que o cliente consiga entender o funcionamento do que se construiu.

#### *Sexta-feira - Testar*

Também chamada de validação, essa é a última fase do DS. Nela o protótipo é apresentado para cinco usuários, os quais serão entrevistados por algum participante do time de desenvolvimento. O conteúdo das perguntas da entrevista terá como foco o uso do protótipo. É recomendável que a entrevista seja feita em uma sala com câmeras discretas para realizar a gravação do uso do protótipo e faça a transmissão em tempo real para os outros participantes do time. Tomadas as notas e verificadas as adaptações necessárias, novas *sprints* podem ser planejadas com a intenção de refinar o protótipo até que ele combine com as necessidades dos clientes, mesmo que as novas iterações da *sprint* não durem cinco dias novamente.

### **Google Design Sprint na educação**

Embora um dos maiores benefícios da *Google Design Sprint* seja o aprendizado da equipe antes de um desperdício de recursos [15], seu uso em sala de aula não é comum, diferente do *Design Thinking* que conta com diversos materiais para este domínio [16], [17]. Muito disso deve se dar por conta da dificuldade de se separar cinco dias de uma semana para trabalho exclusivo em um projeto, quando alunos e professores normalmente se dividem entre muitas disciplinas, cada qual com suas ementas e objetivos de ensino diferentes.

A maior parte das aplicações da *Google Design Sprint* para fins educacionais não foi usada como fator de aprendizado para disciplinas tradicionais [7], [18], sendo bem mais aproveitada na forma de workshops ou cursos completos [19], [20].

### **Google Design Sprint como ferramenta de aprendizagem ativa**

Ser professor em uma classe de Engenharia de *Software* ou qualquer das suas ramificações nos dias atuais pode ser uma experiência desafiadora. O atual cenário lança às salas das universidades alunos que já nasceram em contato com a tecnologia e que pode ficar (e frequentemente fica) facilmente entediado com as aulas tradicionais baseadas em preleção, o que é naturalmente potencializado por causa do perfil tecnológico destes alunos.

Esta geração, que é comumente chamada de *millennial*, possui uma série de características que os tornam um grupo especial de alunos [21], [22], o que faz com que o modelo de aprendizagem tradicional, onde o professor é o centro da sala de aula, não seja tão adequado quanto os modelos de aprendizagem ativa.

No campo da Engenharia de *Software*, práticas como a gamificação, a aprendizagem baseada em projeto, a aprendizagem baseada em problemas e a aprendizagem baseada em times têm sido considerados como boas alternativas ao ensino com base em preleção, apresentando resultados promissores [23]–[27], apoiados pela adoção do modelo de sala de aula invertida que leva o consumo de conteúdos para fora da sala, abrindo espaço para que a aprendizagem ativa ocorra durante os encontros entre professor e alunos, [28] já que dentro do contexto acadêmico, há ementas que não podem ser ignoradas.

Alguns desses modelos, como no caso da aprendizagem baseada em projetos, permitem um encaixe da *Google Design Sprint*, tendo em vista o caráter leve, orientado a objetivos, colaborativo e com foco em resolução de problemas que eles possuem [29]. A semelhança entre objetivos é particularmente forte quando se nota a procura pela construção de artefatos prototípicos tanto em um quanto em outro. [1], [29].

Haja visto os benefícios de ter os graduandos pensando como designers, algumas das maiores universidades dos Estados Unidos e também da Austrália têm se movimentado em direção a adoção do aprendizado sustentado pelo pensamento projetual [7], [8], que também apresentam uma forte consideração de dinâmicas práticas e autênticas no processo de geração de aprendizagem, portanto, a *Google Design Sprint* não está tão distante do contexto educacional que não possa ser usada como ferramenta para aperfeiçoamento de habilidades de inovação, cooperação, pensamento crítico e resolução de problemas, que é uma carência dos estudantes percebida pela indústria [6].

## PREPARAÇÃO DA PESQUISA

O aprendizado rápido proporcionado pela *Google Design Sprint* [14], mesmo que no contexto de uma solução, pode ser um ganho para os alunos de cursos superiores de computação, pois os fazem pensar em termo de inovação para problemas reais. Com esta premissa em mente, construímos um objetivo: obter dados que ajudassem a formular hipóteses sobre como poderíamos construir um modelo de aprendizagem baseado nas técnicas de UX.

A formatação de técnicas dentro da *Google Design Sprint* foi um bom ponto de partida, a partir do qual pudemos adaptar uma sessão mínima que foi executada em uma reunião semanal. Optou-se, portanto, pela modalidade de pesquisa exploratória com estudo bibliográfico e estudo de caso que, quando aplicada, fornecesse dados para serem utilizados na

construção de um modelo de ensino prático de *Design Thinking* nos cursos superiores das universidades brasileiras.

Com base na limitação de tempo, o desafio foi fazer a *Google Design Sprint* de cinco dias em caber em um encontro semanal de modo que seus principais objetivos ainda fossem atingidos. Nesse contexto, os estágios de mapeamento de problemas, de esboço de soluções, de tomada de decisões, de prototipação e de testes tiveram que ser reduzidos de horas para minutos. Não desfigurá-los exigia que as principais técnicas de UX em cada uma delas fossem identificadas, selecionadas e adaptadas. Após um debate entre nós e um intenso refinamento do documento que decidimos chamar de "agenda", obtivemos a estrutura, conforme apresentado na Tabela 1.

- **Pré-Sprint**
  - Escolha do desafio da Sprint
  - Coleta e abstração do fluxo do negócio
  - Convite aos especialistas
- **Entender**
  - – 10 min - Contextualização
  - – Palestras relâmpago sobre os desafios e problemas
    - 5 min - Palestra relâmpago do especialista
    - 5 min - Palestra relâmpago do aluno
  - 10 min - Mapeamento do processo
  - How Might We (HMW)
    - 5 min - Explicação
    - 10 min - Escrita HMW
    - 5 min - Votação silenciosa
    - 5 min - Ordenação dos HMW
- **Buscar soluções / Desenhar**
  - Crazy 8's
    - 5 min - Explicação
    - 5 min - Desenho rápido das oito variações
    - 10 min - Esboço final de uma solução
- **Intervalo**
- **Decidir**
  - Votação silenciosa
    - 5 min - Explicação
    - 30 min - Processo de decisão
- **Prototipar**
- **Pós-Sprint**
  - Validar
  - Responder o questionário

Tabela 1 . Agenda da *Google Design Sprint* adaptada

Antes do dia da *sprint*, pedimos aos alunos que listassem vários problemas pelos quais eles passavam e que quisessem propor uma solução. Depois de mencionar alguns e passar por um sistema de votação, eles escolheram o problema do estágio não obrigatório para o TADS. A partir dele, definimos como meta da Sprint: "*Mais alunos da TADS fazendo estágio*". No modelo original da *Google Design Sprint*, este passo está dentro da *sprint* e deve ser feito pelo

time, no entanto, o fizemos antes para economizar tempo. Em outro momento, ainda antes do dia da *sprint* e também motivado pela economia de tempo, coletamos informações sobre o fluxo atual do processo de estágio no campus Formosa (AUTOR, 2018). Com esse fluxo conhecido, nós o abstraímos até se tornar um fluxo no padrão do mapa de segunda-feira, com cinco etapas que partia das principais partes interessadas até o objetivo final.

Os materiais utilizados na aplicação da *sprint* foram dois pacotes de notas autoadesivas contendo 100 unidades cada, 100 adesivos de bolinha pequenos, 50 adesivos de bolinha maiores, 100 folhas A4, 15 canetas esferográficas, uma sala com uma mesa, giz, borracha, quadro-negro, cadeiras, um notebook, salgados e refrigerante para o intervalo.

## ESTUDO DE CASO: RESULTADOS E DISCUSSÃO

Quinze pessoas participaram da *Sprint*. Lidar com essa quantidade de pessoas foi um desafio, já que o modelo tradicional da *Google Design Sprint* sugere que um máximo de sete pessoas com visões diferentes participem de sua aplicação [1]. No entanto, como o propósito da pesquisa era avaliar a Design Sprint como ferramenta de ensino em sala de aula, adaptável à abordagem de Aprendizagem Baseada em Projetos [29], foi necessário avaliar seu comportamento com um time maior. A falta de heterogeneidade de uma turma que possui uma visão limitada do problema foi resolvida com a presença de um funcionário da COSIE-E que falou em uma das *lightning talks*, e do próprio facilitador, que era o professor de disciplina. Verificamos que a proporção de alunos, especialista e professor pode se transformar em um problema na construção do protótipo da solução. Em futuras iterações desta pesquisa, daremos maior poder aos votos dos envolvidos que estão em minoria, tornando um deles um *Definidor*.

Nos minutos iniciais, fizemos as palestras. O especialista não utilizou os cinco minutos reservados para ele. Além disso, houve detalhes relevantes não mencionados durante seu discurso. Acreditamos que isso influenciou a ideação dos alunos, uma vez que um dos pontos não mencionados foi a burocracia documental que, não por acaso, não recebeu votos suficientes nos cartões HMW para avançar aos estágios de solução propostos no protótipo. Este problema de burocracia na documentação foi identificado como ausente por um dos usuários finais durante o estágio de validação.

Estas foram as considerações do especialista: (1) O estágio não é obrigatório, portanto, a empresa deve pagar o estagiário; (2) Há uma falta de contato com o Instituto de Integração Empresa-Escola (CIEE) e com o Instituto Euvaldo Lodi (IEL) pelo IFG e estudantes.

Soubemos que o especialista tinha problemas em falar em público. Acreditamos que um especialista que não possui inibições dessa natureza possa trazer melhores resultados para a *sprint*.

A palestra dos alunos foi projetada inicialmente para ser feita por apenas um deles, mas eles pediram que qualquer pessoa que quisesse falar tivesse a palavra, que foi concedida pelo

facilitador. A palestra ocorreu como um pequeno debate de 5 minutos, onde todos puderam conversar, expondo diferentes partes do problema. Eles não falaram um no discurso do outro e o tempo foi respeitado. No final, a estratégia funcionou bem e o problema foi muito bem caracterizado

pela visão deles. O problema principal pode ter sido a falta de proporcionalidade entre essa visão e a visão do especialista, como foi discutido anteriormente neste artigo.

Estas foram as considerações dos alunos: (1) Os estagiários são contratados para realizar trabalhos que não estão diretamente relacionados ao curso; (2) Os empregadores exigem conhecimentos avançados que os formandos ainda não obtiveram no curso; (3) As empresas não conhecem as especificidades do curso e pensam que o curso de informática prepara os alunos para trabalhar exclusivamente com infraestrutura; (4) Empresas de desenvolvimento são escassas em Formosa; (5) Há uma falta de estrutura nas empresas existentes, que são pequenas e carecem de demanda para o desenvolvimento de *softwares* maiores; (6) Não há conexão do IFG com o Site Nacional de Emprego (SINE); (7) As empresas desconhecem a existência do curso TADS na cidade; (8) O IFG não conhece empresas que possam oferecer estágio.

De todas as notas desta etapa, vemos os seguintes problemas a serem abordados no *sprint*: **É possível resolver o discurso de interesse das empresas pelos estagiários do TADS, mesmo que tenham que pagá-lo? É possível melhorar a comunicação entre o IFG e as empresas?**

Na execução da dinâmica do *Como Poderíamos (How Might We)* recebemos 53 cartões com perguntas do tipo *How Might We*, dos quais 39 não receberam votos, 7 receberam um voto, 4 receberam dois votos, 1 recebeu três votos, 1 recebeu cinco votos e 1 recebeu seis votos. Os cartões que não receberam votos ou receberam apenas 1 voto foram desconsiderados para as etapas seguintes. A justificativa é que, como cada participante podia votar em seu próprio cartão, provavelmente este recebia votos dos próprios autores ou eram tão irrelevantes para os outros que deveriam ser postos de lado e recuperados em iterações subsequentes, se necessário. A Tabela 2 apresenta o conteúdo dos cartões considerados para a construção da solução. Às vezes, durante a votação, foi necessário que o facilitador solicitasse aos alunos que ficassem em silêncio para que não enviassem os outros membros do time. A propósito, a ansiedade em falar quando deveriam ficar em silêncio foi um problema recorrente.

Na aplicação do exercício *Crazy 8's*, notamos que o tempo de 5 minutos gasto na *sprint* não foi suficiente. Os estudantes alegaram ter entendido o funcionamento da técnica apenas quando o temporizador já havia sido acionado. No entanto, mesmo aqueles que disseram ter entendido antes do início do cronograma, não conseguiram fazer os oito esboços.

O processo de decisão começou após o intervalo. Devido às recomendações de intervalos nas seções da *sprint* [1] e aos intervalos naturais que ocorrem durante as aulas, decidimos adaptar um período de descanso com lanche patrocinado por nós aos participantes. A intenção era que eles se reunissem, conversassem sobre a *sprint* e voltassem para a sala sem se atrasar. Uma conversa sobre a *sprint* aconteceu entre dois

alunos e o facilitador, mas a tentativa de recuperá-los no tempo para a sala falhou e a *sprint* reiniciou com um atraso de 15 minutos. Concluímos que não fazer o intervalo traga resultados melhores.

As ideias nascidas da *sprint* culminaram em um protótipo de 14 telas que, se percorridas, levam o usuário a 5 fluxos diferentes. (Figura 3).

<i>How Might We</i>	Conteúdo dos cartões
Cartões com 2 votos	Como poderemos criar uma comunicação entre empresas e estudantes interessados em estágio através de um sistema onde uma empresa procura estagiários e estagiários procuram empresas?
	Como poderemos despertar o interesse das empresas em solicitar estagiários?
	Como poderemos fazer a COSIE-E conhecer as empresas que fornecem estágio para o curso?
	Como poderemos resolver o problema das vagas disponíveis?
Cartões com 3 votos	Como poderemos atrair mais empresas para o IFG para saber como os cursos funcionam e qual a área de foco?
Cartões com 5 votos	Como poderemos integrar o IFG e as empresas para ter uma melhor comunicação?
Cartões com 6 votos	Como poderemos resolver o problema da falta de experiência dos alunos?

Tabela 2. Cartões How Might We

O *Crazy 8's* feito antes do intervalo produziu 4 esboços sem um único voto, 4 esboços com apenas 1 voto único, 1 esboço com 2 votos simples e 2 esboços com 3 votos simples que foram desconsiderados para as fases posteriores. Isso aconteceu devido ao fato de o processo de decisão ter sido feito em duas etapas com aplicação de votos e supervotos, em que somente aquele que recebeu supervotos procedeu ao refinamento. Convém mencionar que dos esboços que receberam apenas 1 voto simples, havia um que fazia uma representação do mundo real, embora o escopo inicial da *sprint* focasse na construção de um aplicativo móvel que atacasse o problema do estágio. No entanto, este esboço foi um dos escolhidos para ser comentado no passo que em cada aluno selecionou um esboço que não fosse de sua autoria para falar sobre ele. Sua interpretação o levou ao grupo entender como um apelo à facilidade de uso que o aplicativo deveria ter, o que acabou influenciando os protótipos em relação a esse aspecto não funcional da aplicação. Isto nos fez pensar que a abertura do escopo durante a aplicação do *Crazy 8's* pode incentivar a criatividade dos participantes, mesmo que a ideia seja limitada, como no caso de nossa aplicação, onde o objetivo era apenas construir uma aplicação.

Os esboços que serviram de base para o protótipo foram todos aqueles que receberam supervotos e outro que, devido ao grande número de votos simples e por decisão da equipe *sprint*, também foi considerado para etapas posteriores. Na Figura 2, está representada a distribuição dos votos simples com os círculos amarelos e dos supervotos com os círculos pretos.

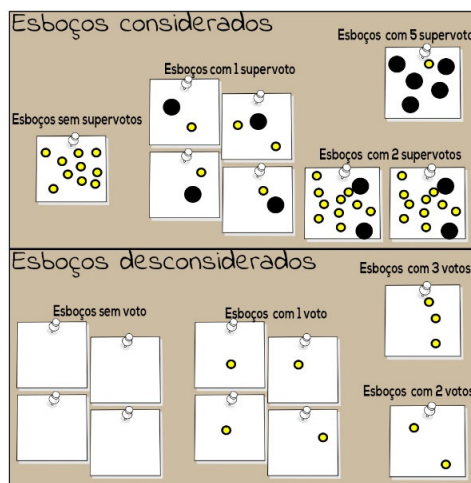


Figura 2. Votos simples e supervotos.

Para ganharmos tempo, separamos os alunos em 4 grupos, onde cada um deveria prototipar uma etapa do fluxo que tinha sido contemplada com as soluções geradas no *crazy 8's* e convenciamos não usar o exercício de storyboard. No entanto, gastamos 46 minutos tentando entender o que tinha sido desenhado e transformando os esboços em um protótipo interativo com o uso do aplicativo POP 2.0 [31].

A tela de *login* se divide entre o fluxo do aluno e o fluxo da empresa, que levam a telas iniciais diferentes (Figura 3). Fazendo login como aluno, a tela inicial mostra botões onde se escolhe o curso no qual o aluno está matriculado.

Na tela inicial do aluno há um menu com link para uma seção contendo vídeo aulas e arquivos de texto sobre conhecimentos específicos que são cobrados nos ambientes de estágio (Figura 3). Esta tela surgiu do HMW mais votado que tratava do problema da falta de experiência dos alunos,



b) como um dos problemas relatados na validação era o fato de as empresas que potencialmente se interessariam pelos serviços dos alunos não saber a finalidade do curso de análise e desenvolvimento de sistemas no IFG ou nem saber que existe oferta de mão de obra de desenvolvedores e analistas de sistemas, o aplicativo poderia gerar postagens automáticas nas redes sociais com a oferta de serviços para que houvesse uma maior divulgação dos estagiários e de suas habilidades;

c) sabendo que grande parte das empresas querem estagiários do curso para trabalhos de infraestrutura, suporte, *help desk* e manutenção, os quais não são especificamente pertencentes à área de conhecimento tratada no curso e que isso acontece por causa de uma falta de conhecimento dos nomes e jargões técnicos da área, o aplicativo poderia fornecer vídeos explicando o que é cada especialidade fornecida como habilidade de um aluno, afinal, muitas vezes, a empresa nem imagina que sua necessidade pode ser resolvida com alguma ferramenta ou técnica específica aprendida em sala de aula.

O aplicativo desenvolvido permitirá adicionar *tags* com informações das habilidades dos alunos (ex.: Programação em Java, Desenho de Interfaces, Implantação de banco de dados, Gestão de Projetos, etc.) e as empresas poderão cadastrar suas necessidades (ex.: Desenvolver um *site*, fazer mapeamento de processos, dar manutenção em um sistema legado, etc.). Nisto, o aplicativo deverá ligar as habilidades dos alunos às necessidades correspondentes.

### Questionário

O questionário com perguntas qualitativas sobre a percepção dos alunos em relação à adaptação da *Google Design Sprint* foi disponibilizado no dia seguinte e ficou disponível por vinte e quatro horas [32]. Para garantir a confiabilidade das respostas, as perguntas foram anônimas. O questionário foi respondido por treze alunos.

O interesse por linguagem de programação deste grupo de alunos é maior que o dobro em relação à análise de sistemas e computação gráfica, etapas comumente ligadas aos aspectos visuais do sistema, conforme apresentado na Figura 4. Se somadas as duas áreas, elas são apenas 80% do total do índice de interesse por linguagem de programação. Isso pode indicar sinais de uma preocupação desequilibrada entre qualidade de código e qualidade dos aspectos visuais, que geralmente é o que os usuários finais levam em consideração.



Figura 4. Áreas de interesse dos alunos.

Os alunos relataram terem aprendido e terem percebido aprendizado nos colegas durante a execução da *Sprint* (Figura, embora não tenham sido tão veementes em relação ao comprometimento do time, conforme apresentado na Figura 5. Suas respostas em relação ao aprendizado não podem ser encaradas como comprovação final de que realmente aprenderam, pois este não era o objetivo final do estudo de caso, mas devem indicar como eles se sentiram em relação ao próprio aprendizado. No máximo, precisávamos avaliar empiricamente se os alunos compreendiam a mecânica da dinâmica enquanto as ações aconteciam. Não faria sentido cobrar deles conhecimentos teóricos sobre a *Google Design Sprint*, se nossa versão é um modelo diferente do usado na indústria.

Para a maioria deles, o tempo foi reconhecidamente um fator limitante, conforme apresentado na Figura 6. A sugestão geral é a de que algo entre doze horas e três dias fosse melhor, o que é coerente, afinal, aliando as duas opções mais escolhidas poderíamos ter, por exemplo, quatro horas divididas em três dias.

No geral, os alunos concordaram sobre a capacidade da *Google Design Sprint* de resolver problemas de forma criativa e manifestaram unânime interesse em participar de novas aplicações da *Sprint* com propósitos educacionais, conforme apresentado na Figura 7, o que representa um bom resultado, pois indica que foi uma experiência agradável aos alunos e pode ter uma boa receptividade também em próximas iterações da *Sprint*.

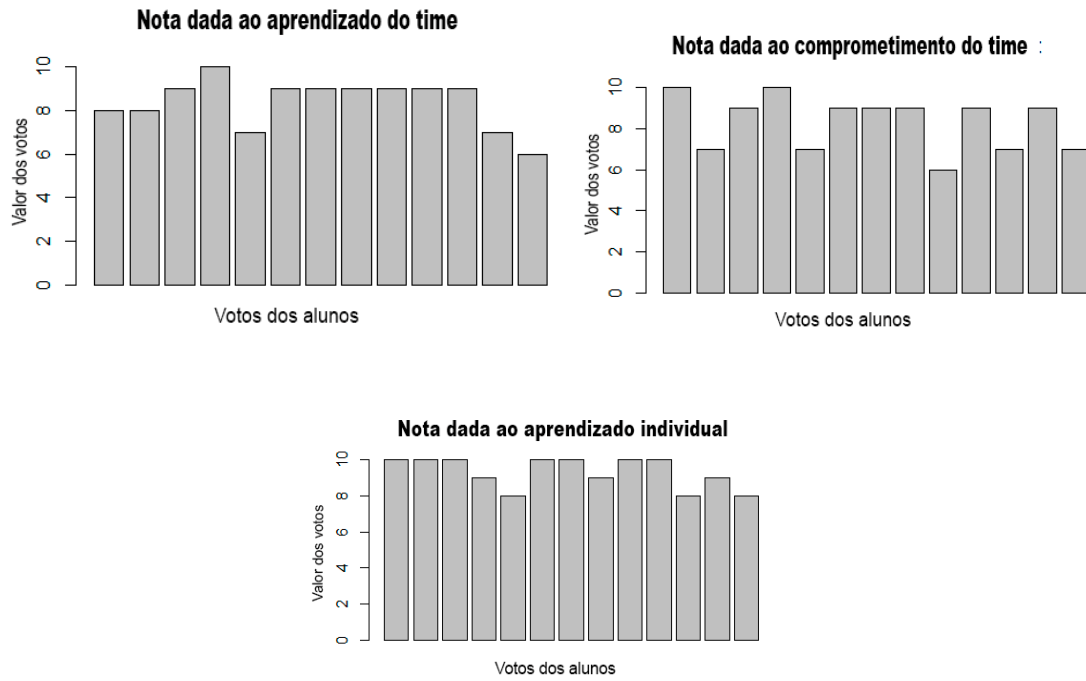


Figura 5. Análise do aprendizado e comprometimento dos alunos.

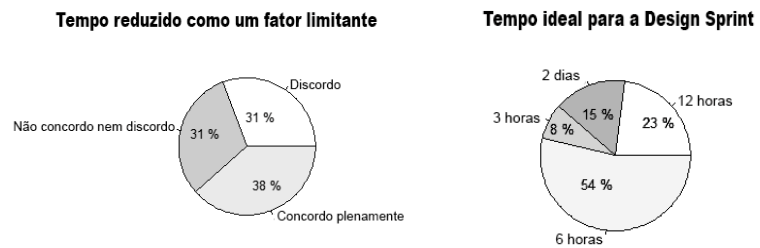


Figura 6. Avaliações relacionadas ao tempo do experimento.

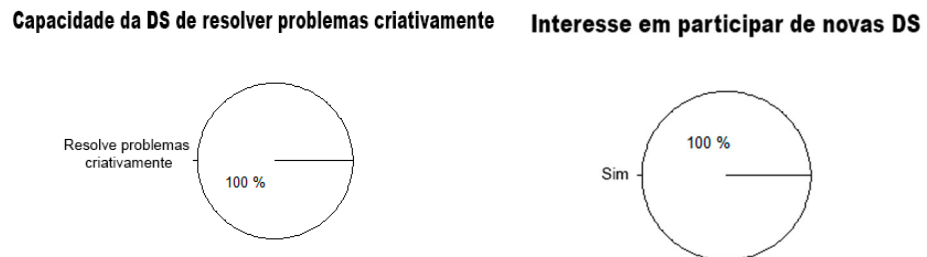


Figura 7. Avaliação geral da execução da *Google Design Sprint*.

## CONCLUSÕES

Este trabalho apresentou o relato da experiência do estudo e aplicação de uma *Google Design Sprint* adaptada para fins educacionais. Os artefatos gerados durante a pesquisa se encontram disponíveis em repositório digital [30,32]. Por ser uma técnica extremamente dinâmica e multidisciplinar, a possibilidade de sucesso em aplicações no contexto das aulas de uma disciplina de engenharia de software é importante.

Alguns estouros de prazo, como o da prototipação e o do intervalo, foram cruciais para que verificássemos que não é viável fazer uma *Google Design Sprint* em tão pouco tempo. Com base nisso, concluímos que o principal cuidado a ser tomado é em relação ao tempo. O experimento nos permitiu identificar que:

- O uso da *Google Design Sprint* é adequado para fornecer um direcionamento para os insights no desenvolvimento de produtos de software utilizando a abordagem de aprendizagem baseada em projetos.
- Os estudantes consideraram a experiência satisfatória na participação e aprendizado em sala de aula no formato de *Google Design Sprint*.
- A metodologia *Google Design Sprint* aplicada em um cenário educacional necessita de um período maior para o seu aprendizado e fixação.
- Os resultados obtidos com o experimento permitiram identificar a aceitação do uso da metodologia pelos estudantes, o que contribui com um aprendizado colaborativo e prático em uma disciplina teórica que ensina a construção de software reais.

Essas considerações serão verificadas nos próximos experimentos utilizando a metodologia *Google Design Sprint* em sala de aula, com os devidos ajustes identificados neste experimento. Este trabalho permitiu identificarmos esses ajustes e remodelar a proposta de uso da metodologia.

Como trabalhos futuros pretendemos testar a *Google Design Sprint* em um cenário de aprendizagem baseada em projetos, utilizando um semestre letivo inteiro, dividindo-a em várias aulas, com o objetivo de verificar se o protótipo produzido em uma *Sprint* dessa natureza tem condições de se tornar um software real e bem avaliado por seus usuários finais. Além disso, pretendemos aplicar a metodologia em outras disciplinas do curso.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] J. Knapp, J. Zeratsky, and B. Kowitz, *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. New York, NY, USA: Simon and Schuster, 2016.
- [2] E. Ries, *A Startup enxuta: como os empreendedores atuais utilizam a inovação contínua para criar empresas extremamente bem-sucedidas*, vol. 1. São Paulo, SP: Leya, 2012.
- [3] A. Mishra, N. Ercil Cagiltay, and O. Kilic, “Software engineering education: some important dimensions,” *Eur. J. Eng. Educ.*, 2007.
- [4] H. Partridge and G. Hallam, “Educating the Millennial Generation for evidence based information practice,” *Libr. Hi Tech*, 2006.
- [5] S. Heggen and C. Myers, “Hiring Millennial Students as Software Engineers,” no. June, 2018.
- [6] A. M. Moreno, M. I. Sanchez-Segura, F. Medina-Dominguez, and L. Carvajal, “Balancing software engineering education and industrial needs,” *J. Syst. Softw.*, vol. 85, no. 7, pp. 1607–1620, 2012.
- [7] D. Hardy, T. Myers, and M. Sankupellay, “Cohorts and Cultures : Developing Future Design Thinkers,” pp. 9–16, 2018.
- [8] N. Anderson, “Achieving higher education graduate attributes in the area of creativity, innovation and problem solving through the use of design thinking,” *QSAApple Conf. Proc.*, pp. 29–33, 2012.
- [9] J. Desconsi, “Design Thinking como um Conjunto de Procedimentos para a Geração da Inovação: Um Estudo de Caso do Projeto G3,” 2012.
- [10] T. Brown, *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*. New York, NY, USA: HarperCollins, 2008.
- [11] J. Vechakul, B. P. Shrimali, and J. S. Sandhu, “Human-Centered Design as an Approach for Place-Based Innovation in Public Health: A Case Study from Oakland, California,” *Matern. Child Health J.*, vol. 19, no. 12, pp. 2552–2559, 2015.
- [12] S. Kieffer, A. Ghouti, and B. Macq, “The Agile UX Development Lifecycle: Combining Formative Usability and Agile Methods,” pp. 577–586, 2017.
- [13] J. Bennedsen and O. Eriksen, “Categorizing pedagogical patterns by teaching activities and pedagogical values,” *Comput. Sci. Educ.*, 2006.
- [14] N. Richter, T. Schildhauer, and P. Jackson, “Meeting the Innovation Challenge: Agile Processes for Established Organisations,” *Entrep. Innov. Leadersh.*, pp. 109–121, 2018.
- [15] R. Banfield, C. T. Lombardo, and T. Wax, *Design Sprint: A Practical Guidebook for Building Great Digital Products*, 1st ed. Sebastopol: O’Reilly Media, 2016.
- [16] J. H. L. Koh, C. S. Chai, B. Wong, and H.-Y. Hong, *Design Thinking for Education*. 2015.

- [17] C. C. Cavalcanti and A. Filatro, *Design thinking na educação presencial, a distância e corporativa*. São Paulo: Saraiva, 2016.
- [18] Y. Wang, “Design sprint concept and VR technology used in international foreign aid training course of textile technology,” *Bol. Tec. Bull.*, vol. 55, no. 17, pp. 335–341, 2017.
- [19] E. Sari and A. Tedjasaputra, “Designing Valuable Products with Design Sprint Eunice!,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10516 LNCS, pp. 515–517, 2017.
- [20] L. Valentine, T. Kroll, F. Bruce, C. Lim, and R. Mountain, “Design Thinking for Social Innovation in Health Care,” *Des. J.*, vol. 20, no. 6, pp. 755–774, 2017.
- [21] A. Loureiro and I. Messias, “Competences and Learning Profiles of Digital Age ’ s Students,” *Engag. Digit. ANtives High. Educ. Settings*, 2016.
- [22] J. L. Frand, “The information age mindset: Changes in students and implications for higher education,” *Educ. Rev.*, vol. 35, no. October 2000, pp. 15–24, 2000.
- [23] S. Matalonga, G. Mousqués, and A. Bia, “Deploying Team-based Learning at Undergraduate Software Engineering Courses,” in *Proceedings of the 1st International Workshop on Software Engineering Curricula for Millennials*, 2017, pp. 9–15.
- [24] M. M. Alhammad and A. M. Moreno, “Gamification in Software Engineering: A systematic mapping,” *J. Syst. Softw.*, 2018.
- [25] A. Karabulut-Ilgu, N. Jaramillo Cherez, and C. T. Jahren, “A systematic review of research on the flipped learning method in engineering education,” *Br. J. Educ. Technol.*, vol. 49, no. 3, pp. 398–411, 2018.
- [26] M. L. Fioravanti, B. Sena, L. N. Paschoal, L. R. Silva, A. P. Allian, E. Y. Nakagawa, S. R. S. Souza, S. Isotani, and E. F. Barbosa, “Integrating Project Based Learning and Project Management for Software Engineering Teaching: An Experience Report,” in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 2018, no. Icmc, pp. 806–811.
- [27] P. Dolog, L. L. Thomsen, and B. Thomsen, “Assessing Problem-Based Learning in a Software Engineering Curriculum Using Bloom’s Taxonomy and the IEEE Software Engineering Body of Knowledge,” *ACM Trans. Comput. Educ.*, vol. 16, no. 3, pp. 1–41, 2016.
- [28] A. Sams and O. Washington, *Flip your classroom*. Eugene, Oregon: International Society for Technology in Education, 2012.
- [29] W. N. Bender, *Project-based learning: Differentiating instruction for the 21st century*. Thousand Oaks, CA: Corwin Press, 2012.
- [30] V. G. Ferreira, “Fluxograma atual de estágio no IFG,” 2018. [Online]. Available: <https://doi.org/10.6084/m9.figshare.6223787>. [Accessed: 19-Jul-2018].
- [31] WOOMO, “POP 2.0 - Prototyping on paper.” 2017.
- [32] V. G. Ferreira, “Questionário da Design Sprint,” 2018. [Online]. Available: <https://doi.org/10.6084/m9.figshare.6223790>. [Accessed: 20-Jul-2018].