

UNIVERSIDADE EVANGÉLICA DE GOIÁS - UNIEVANGÉLICA
ENGENHARIA DE COMPUTAÇÃO/ENGENHARIA DE SOFTWARE

LEONARDO DUARTE AMORIM

Implementação de um *Framework* para Identificação de Valor do Produto em
Fábrica de *Software Acadêmica*

Anápolis
Dezembro, 2021

UNIVERSIDADE EVANGÉLICA DE GOIÁS - UNIEVANGÉLICA
ENGENHARIA DE COMPUTAÇÃO/ENGENHARIA DE SOFTWARE

LEONARDO DUARTE AMORIM

Implementação de um *Framework* para identificação de valor do produto em
fábrica de *software* acadêmica

Trabalho apresentado ao Curso de Engenharia de Computação da Universidade Evangélica de Goiás – UniEVANGÉLICA, da cidade de Anápolis-GO como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Orientador (a): Prof. Ms. Walquíria Fernandes Marins

Anápolis
Dezembro, 2021

UNIVERSIDADE EVANGÉLICA DE GOIÁS - UNIEVANGÉLICA
ENGENHARIA DE COMPUTAÇÃO/ENGENHARIA DE SOFTWARE

LEONARDO DUARTE AMORIM

Implementação de um *Framework* para identificação de valor do produto em
fábrica de *software* acadêmica

Monografia apresentada para Trabalho de Conclusão de Curso de Engenharia de Computação da Universidade Evangélica de Goiás - UniEVANGÉLICA, da cidade de Anápolis-GO como requisito parcial para obtenção do grau de Engenheiro(a) de Computação.

Aprovado por:

**Walquíria Fernandes Marins, Mestre, UniEVANGÉLICA
(ORIENTADOR)**

**Natasha Sophie Pereira, Doutora, UniEVANGÉLICA
(AVALIADOR)**

**Alexandre Tannus, Mestre, UniEVANGÉLICA
(AVALIADOR)**

Anápolis, 08 de dezembro de 2021.

FICHA CATALOGRÁFICA

AMORIM, Leonardo Duarte. **Implementação de um *framework* para identificação de valor do produto em fábrica de *software* acadêmica.** Anápolis, 2021. (Universidade Evangélica de Goiás – UniEVANGÉLICA, Engenheiro(a) de Computação, 2021).

Monografia. Universidade Evangélica de Goiás, Curso de Engenharia de Computação, da cidade de Anápolis-GO.

1. Identificar valor do produto. Fábrica de *software* acadêmica. Processo de desenvolvimento de *software*.

REFERÊNCIA BIBLIOGRÁFICA

AMORIM, Leonardo Duarte. **Implementação de um *framework* para identificação de valor do produto em fábrica de *software* acadêmica.** Anápolis, 2021. 67 p. Monografia - Curso de Engenharia de Computação Universidade Evangélica de Goiás - UniEVANGÉLICA.

CESSÃO DE DIREITOS

NOMES DOS AUTORES: Leonardo Duarte Amorim.

TÍTULO DO TRABALHO: Implementação de um *framework* para identificação de valor do produto em fábrica de *software* acadêmica.

GRAU/ANO: Graduação /2021.

É concedida à Universidade Evangélica de Goiás - UniEVANGÉLICA, permissão para reproduzir cópias deste trabalho, emprestar ou vender tais cópias para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte deste trabalho pode ser reproduzida sem a autorização por escrito do autor.

Leonardo Duarte Amorim

Anápolis, 08 de dezembro de 2021

AGRADECIMENTOS

Em primeiro lugar, quero agradecer à Deus, que fez com que meus objetivos fossem alcançados, durante todos os meus anos de estudos e por ter me dado saúde e coragem para conseguir concluir mais uma etapa da minha vida.

Quero também agradecer minha família por todo apoio e incentivo e também a todos os amigos, mas em especial Vinícius Moreira e Rafaela Batista, que estavam sempre ao meu lado me orientando e ajudando em questões pessoais, profissionais e acadêmicas.

À professora Walquíria Marins, por ter sido meu orientador e ter desempenhado tal função com dedicação, amizade e paciência, pois não foram caminhos fáceis até chegar este momento.

RESUMO

A identificação de valor de um produto de *software* é um problema recorrente em diversas empresas de tecnologia e, quando não entende-se a necessidade do cliente, a tendência da empresa é investir de outras formas para conquistar seus clientes sendo que sua fidelização seria mais eficiente caso entendessem sua real necessidade. O objetivo central desse trabalho é identificar uma abordagem para descoberta de valor do produto para o cliente em um processo de desenvolvimento de *software* no ambiente de uma fábrica de *software* acadêmica, a Fábrica de Tecnologia Turing (FTT) e aplicar essa abordagem no ambiente escolhido. Através de uma análise bibliográfica sobre o processo base exigido para desenvolvimento de um produto de *software* para a engenharia de *software*, sobre as metodologias ágeis já utilizadas no ambiente da fábrica de *software* acadêmica e sobre os *frameworks fit for purpose* e *design thinking* foi possível comparar as similaridades entre os processos de desenvolvimento proposto por cada metodologia ou *framework* e identificar características que auxiliem na identificação de valor do produto. Com o estudo do processo de desenvolvimento da FTT foi possível definir métricas para acompanhamento das áreas envolvidas no processo de desenvolvimento de *software*, subprocesso para coleta e análise de feedback dos clientes em todas as fases de desenvolvimento (planejamento, implementação e entrega do produto) e identificar falhas no processo em relação a participação do cliente. Após esses estudos possibilitou propor alterações no processo de desenvolvimento na intenção de gerar valor para o produto com o foco tanto no produto gerado quanto na experiência durante o processo de desenvolvimento. Com isso o trabalho destaca a importância da participação do cliente no processo de desenvolvimento de *software*.

Palavras-chave: Identificar valor do produto. Fábrica de *software* acadêmica. Processo de desenvolvimento de *software*. Métricas.

LISTA DE FIGURAS

Figura 1 - Processo <i>SCRUM</i>	24
Figura 2 - Processo <i>OpenUp</i>	26
Figura 3 - Infográfico framework F4P	29
Figura 4 - Comparação do processo base de ES com os processos dos frameworks e metodologias.....	36
Figura 5 - Processo de Desenvolvimento da FTT	39
Figura 6 - Cartão de OKR	41
Figura 7 - Cartão Fit Box Score	43
Figura 8 - Exemplo box score	44
Figura 9 - Cartão de OKR preenchido.....	50
Figura 10 - Resultado final FBS	51

LISTA DE GRÁFICOS

Gráfico 1 - Participação dos clientes nas reuniões de planejamento	46
Gráfico 2 - Participação dos clientes nas reuniões de revisão	47
Gráfico 3 - Participação dos clientes nas reuniões diárias.....	47
Gráfico 4 - Experiência dos colaboradores dentro da FTT.....	48
Gráfico 5 - Quantidade de colaboradores que já trabalharam em outros ambientes	48
Gráfico 6 - Nível de conhecimento dos colaboradores sobre processo de desenvolvimento da FTT	49

LISTA DE TABELAS

Tabela 1 - Cronograma da execução33

LISTA DE ABREVIATURAS E SIGLAS

Siglas	Descrição
DT	<i>Design Thinking</i>
ES	Engenharia de <i>Software</i>
F4P	<i>Fit for Purpose</i>
FBS	<i>Fit Box Score</i>
FS	Fábrica de <i>Software</i>
FSA	Fábrica de <i>Software Acadêmica</i>
FTT	Fábrica de Tecnologias Turing
KPI	<i>Key Performance Indicator</i>
NPS	<i>Net Promoter Score</i>
OKR	<i>Objective and Key Results</i>
PBL	<i>Product Backlog</i>
PO	<i>Product Owner</i>
SM	<i>Scrum Master</i>
WIP	<i>Work in Progress</i>

SUMÁRIO

1. INTRODUÇÃO	16
2. FUNDAMENTAÇÃO TEÓRICA.....	20
1.1 Engenharia de <i>Software</i>	21
1.2 Fábrica de <i>software</i> acadêmica	22
1.3 <i>Frameworks</i> ágeis aplicados na FTT	23
1.3.1 <i>SCRUM</i>	23
1.3.2 <i>OPENUP</i>	25
1.4 <i>Frameworks</i> com foco na identificação e entrega de valor de produto.....	27
1.4.1 <i>Design Thinking</i>	27
1.4.2 Fit For Purpose	28
1.5 Observações finais	30
3. METODOLOGIA DA PESQUISA	32
4. DESENVOLVIMENTO.....	35
3.1 Análise comparativa dos <i>frameworks</i>	35
3.2 Análise do processo de desenvolvimento da FTT	37
3.3 Proposta e aplicação das alterações propostas no processo de desenvolvimento.....	40
3.3.1 Primeira frente – Metrificação do processo de desenvolvimento da FTT	40
3.3.2 Segunda frente – Subprocesso para análise e acompanhamento de feedback do cliente	42
5. RESULTADOS E DISCUSSÕES.....	45
6. CONCLUSÃO E CONSIDERAÇÕES FINAIS.....	52
7. REFERÊNCIAS.....	54
8. APÊNDICES	57
APÊNDICE A – Pesquisa inicial com colaboradores da FTT	57
APÊNDICE B – Apresentação sobre o F4P	60
APÊNDICE C – Cartões preenchidos de FBS e OKR	65

1. INTRODUÇÃO

O sucesso de uma empresa está atrelado aos “3P’s” que são, respectivamente: as pessoas que compõem a empresa, o produto gerado e os processos de desenvolvimento desse produto (LEMONIS, 2015). Esta interpretação, evidencia a necessidade da preocupação com as funções e o bem estar de cada colaborador, entender a competitividade do produto no mercado e se ele se adequa às necessidades do cliente. Além disso, compreender se os processos utilizados são claros para todos os colaboradores e eficientes para garantir que as entregas sejam feitas com qualidade dentro do prazo estabelecido para o projeto (LEMONIS, 2015).

Uma fábrica de *software* (FS) também deve se preocupar com as pessoas, processos e produtos, mas, além disso, devem se atentar a como seus processos se adequam a produção massiva de *softwares* com modelos pré-moldados e como é possível agregar valor a diversos produtos gerados com o processo genérico. Logo, esse cenário de agregar valor ao produto, se torna mais complexo devido à quantidade de *softwares* produzidos e ao processo abrangente (POCIVI, 2011; TEBET, 2020).

Apesar da complexidade, entender o processo de produção de um *software*, em que o produto está alinhado à necessidade do cliente é uma atividade essencial, no entanto esse é um dos maiores desafios enfrentados durante todo o ciclo de vida produto, pois gera uma deficiência do negócio. Com isso, pode-se antecipar a fase de declínio e fazer com que o produto seja descontinuado, ou fazer que o produto não seja aceito pelo mercado, não conseguindo recursos suficientes para sua própria manutenção (PERALTA, 2020).

Embora existam metodologias que auxiliam na perseverança do produto do mercado através da melhora na velocidade de entrega de valor para o cliente como, por exemplo, *Lean Statup*, *Lean Innovation*, *Lean Product* e *Costumer Development*, nenhuma delas se preocupa integralmente na identificação de valor do produto para atender à necessidade do cliente (PERALTA, 2020). Eric Ries (2012, p.16) reforça essa teoria ao afirmar que a metodologia *Lean Statup* se baseia no processo de manufatura e tem como princípios o reaproveitamento das experiências de cada colaborador, redução dos tamanhos dos lotes de produção e produção do tipo *just in time* com maior ênfase no controle de estoque e na aceleração do tempo de ciclo.

Logo, cada uma das metodologias possuem características únicas em seu processo que contribuem de forma isolada nas etapas de desenvolvimento de um produto de *software*.

Sendo assim aplicação de duas ou mais metodologias diferentes no mesmo processo de desenvolvimento de *software* pode gerar dúvidas quanto a sua aplicação e utilização quando a finalidade é a identificação do valor do produto e não a somente a entrega (PERALTA, 2020; SOMMERVILLE, 2011).

Além da dificuldade em identificar o valor do produto em um projeto único, as fábricas de *software* trabalham para criar diversos produtos através de projetos que utilizam um processo genérico, adaptando somente alguns artefatos e rotinas para as necessidades de cada projeto (FERNANDES; TEIXEIRA, 2011). A ocorrência de múltiplos projetos simultâneos pode dificultar ainda mais o processo de identificação do valor do produto, uma vez que nem sempre o cliente consegue se expressar da maneira correta ou a equipe não consegue absorver a expectativa do cliente (RODRIGUES, 2014).

Dada a dificuldade na identificação de valor do produto é questionado se há alguma metodologia ou *framework* com este foco. Caso exista, seria possível adaptar ao processo produtivo de desenvolvimento de *software*? Se não, como as metodologias que já existem podem colaborar para construção de um novo modelo focado na identificação de valor? Quais métricas seria possível implementar para avaliar a adequação do produto à necessidade do cliente? Quais seriam os impactos da sua aplicação em processo?

A ubiquidade de *software* na vida cotidiana é algo indiscutível, uma vez que está presente até mesmo nas atividades diárias mais simples sem precisar de altos investimentos. Dado este fato, afirma-se que 100% da população mundial é afetada diretamente ou indiretamente por tecnologias de *software* sendo que grande parte dessas pessoas tem contato direto com essas tecnologias e, mesmo com a crescente presença de *software* na vida humana, não é dada a devida atenção para desenvolver produtos que forneçam uma boa experiência para o usuário (TORRES, 2015).

De acordo com Salviano (2006), “a comunidade tem procurado entender a natureza das atividades de *software* e sua relação com as outras disciplinas da engenharia, e desenvolver métodos, técnicas, modelos e ferramentas para a engenharia de *software*”, ademais Torres (2015) confirma que o processo de desenvolvimento de *software* teve uma grande evolução, no entanto defende a necessidade de uma área específica para gerenciar o produto com o intuito de desenvolver algo que atenda as expectativas e necessidade do usuário, como prova disso, o *Scrum* surgiu em meados dos anos 2000 e especificou papéis como o de *Product Owner* (PO).

Portanto, entender o propósito do cliente em relação ao produto é essencial para sucesso de qualquer negócio (PERALTA, 2020), para David J. Anderson e Alexei Zheglov (2018, p. 38), atividade que se inicia na Análise de Viabilidade do sistema, segue para a etapa de Engenharia de Requisitos, mas requer acompanhamento e validação durante todo o ciclo de vida (SOMMERVILLE, 2011). A falta de capacidade para entender o propósito é o mesmo que “voar às cegas”, essa deficiência leva os consumidores a procurarem seus concorrentes e, por não conseguirem identificar o porquê eles tomaram essa decisão, as estratégias elaboradas serão falhas e os investimentos feitos de maneira errada.

Logo, “muitas empresas, no intuito de fechar uma venda, gastam centenas de milhares de reais para encantar o cliente, quando muitas vezes o cliente esperava apenas o atendimento de sua necessidade.”, por essa razão entender a necessidade do cliente e colocá-lo em destaque antes de iniciar qualquer projeto pode gerar economias de tempo e dinheiro, aumentar a capacidade produtiva da equipe e satisfação do cliente (RODRIGUES, 2014, p.74).

O objetivo geral desse trabalho é identificar uma abordagem para descoberta de valor do produto para o cliente em um processo de desenvolvimento de *software*. Logo, foi feita uma análise comparativa entre o processo base de desenvolvimento defendido por Sommerville (2011) e outras metodologias ágeis para entender se elas seguiam ou percorriam todo o processo de desenvolvimento e, também, identificar características dessas metodologias voltadas para identificação de valor do produto com objetivo de selecionar uma para aplicar no processo de desenvolvimento da FTT.

No intuito de aplicar um subprocesso para identificação de valor do produto no processo de desenvolvimento de *software*, foi selecionado o ambiente acadêmico da Fábrica de Tecnologia Turing (FTT). Com o ambiente selecionado era esperada que fosse aplicado uma pesquisa para mapear a situação atual da FTT e também uma análise do processo de desenvolvimento, esse estudo e análise das metodologias permitiram propor alterações do atual processo, inclusive com a inserção de métricas para acompanhamento da evolução do produto, equipe e objetivos do projeto.

Logo, a pesquisa se divide em sessões, sendo a primeira a fundamentação teórica onde será introduzido todo o material de apoio para a execução da pesquisa, a segunda sessão é a de desenvolvimento em que é explicado como o trabalho foi realizado e aplicada no ambiente da FTT, a terceira tem o foco em apresentar e discutir os resultados do trabalho aplicado e,

por último, a conclusão em que é analisado se a problemática levantada foi solucionada através dos objetivos definidos para a pesquisa.

2. FUNDAMENTAÇÃO TEÓRICA

Um produto pode ser definido de diversas formas para diferentes propósitos, no entanto Eric Ries (2012, p.25), em seu livro *A startup enxuta*, apresenta uma definição muito completa e abrangente, logo ele usa a definição de produto como:

“... aquela que abrange qualquer fonte de valor para as pessoas que se tornam clientes. Qualquer coisa que os clientes vivenciam da interação com uma empresa deve ser considerada parte do produto daquela empresa. Isso é verdade em relação a uma quitanda, um site de comércio eletrônico, um serviço de consultoria e uma entidade de serviço sem fins lucrativos.”

Após entender a definição de um produto e relacioná-la com o contexto de desenvolvimento de *software* surge o conceito de produto digital que para Joaquim Torres (2020, p.34) é “qualquer *software* que tenha usuários”, para ele é possível classificar um produto como digital observando tipo de público que ele atende e pela forma que ele é entregue aos consumidores finais.

Produtos digitais podem estar presentes em diversos tipos de empreendimento, mas é necessário entender a natureza de atuação da empresa para explicitar a função do produto no negócio, sendo assim há a natureza digital em que o produto de *software* é a parte mais importante do negócio, sem ele o negócio não existiria, há a natureza tradicional em que o produto de *software* potencializa o crescimento de algo que já existia e não representa a parte principal do negócio e, por último, há a natureza tradicional nascida digital em que produto foi concebido como um *software*, porém ele poderia existir fisicamente, neste caso também não representa a parte principal do negócio (TORRES, 2020).

Compreender o que é um produto e sua função em um negócio é essencial, no entanto, para entender seu comportamento em relação a um negócio, é necessário conceituar o que é valor de produto para o cliente. Sendo assim, Mello e Leão (2008) conceituam o valor de produto para o cliente de três¹ formas distintas, porém para esse trabalho o valor do produto para o cliente deve ser considerado como “uma relação de custo e benefício, ou seja, a diferença entre os valores que o cliente ganha (benefícios funcionais e subjetivos) comprando

¹ Disponível em: <http://www.spell.org.br/documentos/ver/31086/uma-revisao-critica-sobre-o-uso-de-diferentes-conceitos-de-valor-no-marketing/i/pt-br>.

e usando um produto, e os custos (dinheiro, esforço, custo físico ou psíquico) para obter este produto.”

Apesar das diferentes naturezas de atuação de um produto nas empresa o desenvolvimento de um produto deve seguir um processo e, conseqüentemente, esse processo deve ser gerenciado para a garantir que o produto gerado seja manutenível com o decorrer do tempo, confiável em relação aos dados utilizados e gerados por ele, eficiente no que se propôs executar e aceitável pelo seu público alvo (SOMMERVILLE, 2011).

Processos são indispensáveis para desenvolvimento de um produto, no entanto uma dificuldade encontrada em relação a assimilação de processos é que as pessoas se familiarizam mais com processos físicos e os processos de desenvolvimento de um produto são mais intelectuais e organizacionais, consistindo em uma sequência de passos ou atividades para conceber, projetar e comercializar o produto, outras dificuldades encontradas são a rigidez, ausência ou pluralidade de processos.(EPPINGER; ULRICH, 2016).

1.1 Engenharia de *Software*

Entender os processos dos *frameworks* é importante, no entanto é indispensável compreender a disciplina que estuda todas as ações necessárias para o desenvolvimento de *software*, a Engenharia de *Software* (ES), nela podemos compreender as relações entre os processos básicos de desenvolvimento de *software* e como eles podem ser aplicados no cotidiano de uma equipe (SOMMERVILLE, 2011).

A IEEE, citada por Pressman (2011, p.39), define a ES como “a aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de *software*”, sendo assim Sommerville (2011, p.05) complementa essa definição quando afirma que ES “não se preocupa apenas com os processos técnicos do desenvolvimento de *software*. Ela também inclui atividades como gerenciamento de projeto de *software* e desenvolvimento de ferramentas, métodos e teorias para apoiar a produção de *software*”.

A ES define que os processos de desenvolvimento de *software* devem possuir quatro atividades essenciais, sendo elas a especificação de requisitos, projeto e implementação de *software*, validação de *software* e evolução de *software*. Todas essas etapas são essenciais para garantir que o produto gerado possa atingir os requisitos mínimos de manutenibilidade, confiabilidade, eficiência e aceitação por parte do usuário final (SOMMERVILLE, 2011).

A etapa de especificação de requisitos para a ES é o momento do processo em que os colaboradores são responsáveis por entender as necessidades do produto, nela é criado um documento detalhando os aspectos do sistema que será construído e essa etapa deve ser executada antes ou em paralelo com a etapa de projeto e implementação de *software* (PRESSMAN, 2011; SOMMERVILLE, 2011).

A etapa de projeto e implementação de *software* para Somerville (2011) é a fase em que busca definir o planejamento de como será executado o projeto, desenvolvendo artefatos como, por exemplo, documento arquitetural e/ou *roadmaps*, além de desenvolver o que foi levantado pela etapa de especificação de requisitos, gerando assim os incrementos funcionais.

Com os incrementos gerados é possível realizar a validação dos requisitos, isto é, verificar se o que foi especificado está de acordo com o que foi implementado e, a última etapa, deve ser a de evolução do *software* em que o objetivo é a entrega e o acompanhamento do produto gerado com a finalidade de encontrar pontos de melhoria, caso sejam encontrados pontos de melhoria todo o processo deve ser reiniciado, dando início a um processo cíclico e não linear (SOMMERVILLE, 2011).

Indiscutivelmente, manter um processo de desenvolvimento de *software* é trabalhoso, por essa razão um dos elementos essenciais para nortear a evolução constante deste processo são as métricas, que possibilita realizar uma análise quantitativa sobre pontos de atenção ou pontos exemplo para manter a “saúde” do processo, no entanto métricas de *software* são complexas de serem aplicadas devido à grande quantidade de incertezas durante um projeto (PRESSMAN, 2011).

1.2 Fábrica de *software* acadêmica

Para Fernandes e Teixeira (2011) FS é um ambiente voltado para a produção múltipla de produtos de *softwares* com diferentes escopos e natureza, tendo como base um processo estruturado, controlado e melhorado de forma contínua seguindo as boas práticas da engenharia industrial. No entanto, o processo de uma FS não deve tratar uma pessoa como uma máquina automatizada, uma vez que o processo de uma FS é mais criativo e envolvendo talento das pessoas que o compõe (BORGES; MACHADO; CARVALHO, 2011).

Neste contexto de um processo criativo e envolvendo talento das pessoas dentro de uma FS, surge o conceito de Fábrica de *Software Acadêmica* (FSA), em que a finalidade é capacitar os alunos de instituições de ensino superior para o mercado de trabalho através de

desafios, palestras, pesquisas, eventos, cursos e sempre promovendo a interdisciplinaridade (BORGES; MACHADO; CARVALHO, 2011).

Sendo assim, em conjunto com os Cursos de Bacharelados em Computação, a UniEvangélica fundou em 2006 a FTT, uma FSA com foco em desenvolvimento de *softwares* personalizados para a própria instituição ou para instituições parceiras a UniEvangélica e, desde 2006, forma profissionais mais qualificados para o mercado de trabalho, uma vez que o próprio grupo gestor da FTT incentiva a adoção das tecnologias e ferramentas mais atuais do mercado (SOUSA et al., 2018).

1.3 Frameworks ágeis aplicados na FTT

Fernandes e Teixeira (2004), citados por Pocivi (2011), relataram que uma FS deve possuir *frameworks* que contemplem tanto o processo produtivo como o processo gerencial. Logo, a FTT utiliza um processo híbrido baseado nos *frameworks* ágeis *OpenUp* e *Scrum*, sendo o *OpenUp* utilizado para nortear o processo produtivo de desenvolvimento de *software* iterativo e o *Scrum* como processo gerencial (SOUSA et al., 2018).

1.3.1 SCRUM

O *framework Scrum* foi citado pela primeira vez em 1986 por Hirotaka Takeuchi e Ikujiro Nonaka e foi aperfeiçoada por Jeff Sutherland e Ken Schwaber até 1995, ano o qual publicaram o artigo “*Scrum and the perfect storm*”² e, apesar de seu ano de publicação ser anterior à data do manifesto ágil, o *SCRUM* ainda é considerado um *framework* ágil de desenvolvimento de projetos de *software* (PHAM; PHAM, 2012) e, para Camargo (2018), “o método mais usado nos dias de hoje, principalmente porque pode ser integrado a outros métodos ágeis com facilidade, aplicando-se não só ao desenvolvimento de *softwares* como a qualquer ambiente de trabalho”.

O processo *Scrum* deve ser empírico incitando a melhoria contínua através da observação da realidade ou experiência dos colaboradores e também aplica conceitos da metodologia lean thinking³. Além de ser um processo empírico e aplicar a metodologia lean

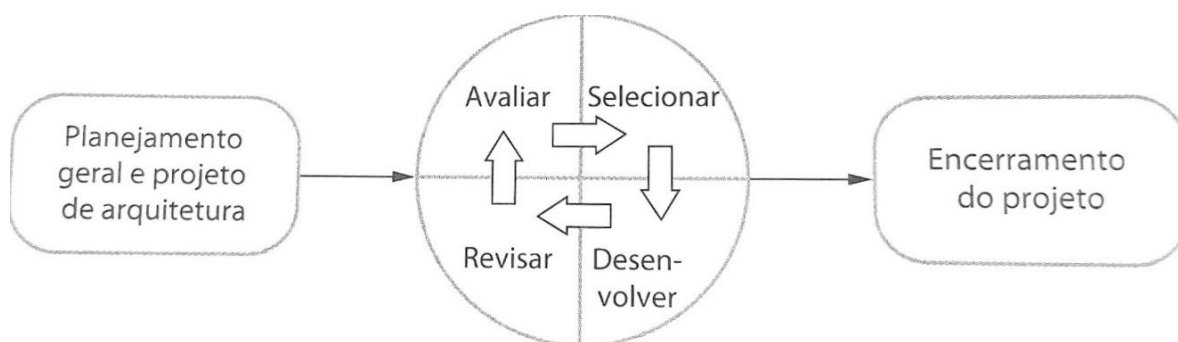
² Disponível em: <http://www.controlchaos.com/my-articles>. Acessado em: 23 de mar. De 2021.

³ Uma metodologia criada com o foco em evitar desperdícios e agregar valor ao produto, inicialmente aplicada em ambiente fabris e que, ao longo dos anos, foi adaptada para outras áreas (WOMACK; JONES, 2004).

thinking, o *Scrum* se baseia em três pilares: Transparência, Adaptação e Inspeção, gerando valores como compromisso, foco, abertura, respeito e coragem (SCHWABER; SUTHERLAND, 2020).

O *SCRUM* é dividido em três fases, conforme a Figura 1, sendo na primeira em que são definidos o planejamento geral do projeto e arquitetura do *software*, a segunda são os ciclos Sprint, em que cada ciclo é desenvolvido um incremento utilizável do *software* seguindo sempre o ciclo iterativo e incremental de avaliar, selecionar, desenvolver e revisar, já a terceira, e última fase, é a fase onde encerra o projeto entregando manuais de sistema, documentação exigida, quadros de ajuda e avalia a lições aprendidas com o projeto (SOMMERVILLE, 2011).

Figura 1 - Processo SCRUM



Fonte: (SOMMERVILLE, 2011).

Além da estrutura inicial do processo pré-definida, o *SCRUM* reconhece os colaboradores como *Scrum* team e os divide em papéis que são: 1) Product Owner: responsável maximizar o valor do produto junto a equipe; 2) *Scrum* Master: que é responsável por manter o bom andamento do projeto conforme o guia do *Scrum*; 3) Developers: são os responsáveis por criar qualquer incremento funcional na sprint (SCHWABER; SUTHERLAND, 2020).

Com a equipe definida será possível realizar uma Sprint (um ciclo iterativo e incremental de desenvolvimento de *software*) que além do tempo de desenvolvimento é composta pelos eventos: 1) Sprint planning: evento em que se define a meta da Sprint, logo nesse momento todo o *Scrum* team e os convidados pelo product owner (PO) discutem quais itens do product backlog (PBL) devem ser inseridos na sprint backlog; 2) Daily *Scrum*; evento em que o *Scrum* team relata qual o progresso em relação a meta da sprint auxiliando assim para a elaboração de um plano de ação para o dia; 3) Sprint review: evento em que os

steakeholdres⁴ realizam a inspeção de todos os artefatos gerados durante a sprint; 4) Sprint retrospective: evento em que são identificados os pontos de atenção em relação a processos, ferramentas (SCHWABER; SUTHERLAND, 2020).

Para Schwaber (2004) e Schwaber e Beedle (2001) *SCRUM* é um metodologia ágil com o foco gerenciamento de desenvolvimento de *software* iterativo (apud. SOMMERVILLE, 2011), logo é uma metodologia que não se aplica a todos os contextos de desenvolvimento de *software* e há a necessidade de adaptar os processos de acordo com ambiente (PHAM; PHAM, 2012)

Por esse motivo a Fábrica de Tecnologias Turing (FTT), uma FSA mantida pela Associação Educativa Evangélica e coordenada pelos cursos bacharelados em computação, além de utilizar a metodologia *Scrum* para o processo gerencial, utiliza-se também o *OpenUp* para o desenvolvimento de *software* iterativo e incremental (POCIVI, 2011; SOUSA et al., 2018).

1.3.2 OPENUP

Pocivi (2011, p.24) defini o *OpenUp* como

“Um processo de desenvolvimento com abordagem ágil, voltado para equipes pequenas. Ele foi criado pela IBM, reflete as boas práticas contidas no RUP e pode ser extensível e otimizado para as necessidades de cada projeto ou ambiente.”.

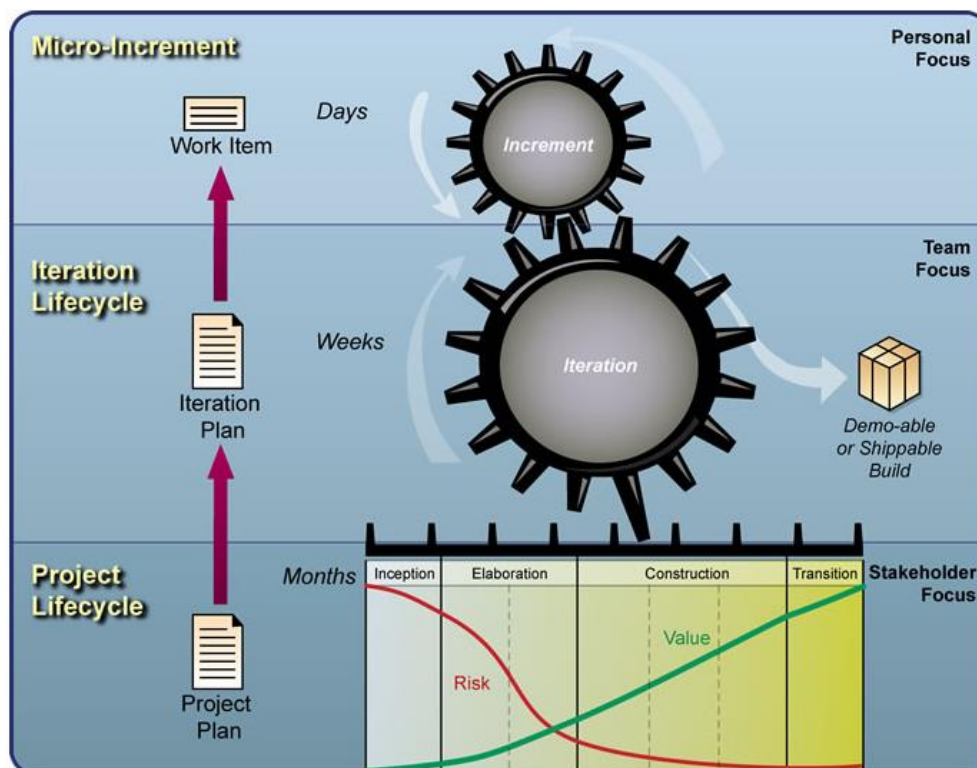
A metodologia tem como características base ser mínima, utilizando somente os artefatos necessários para o desenvolvimento de *software*, completa, abrangendo todos os passos do processo de desenvolvimento ágil de *software*, e extensível, para se adaptar à realidade de cada projeto, podendo também ser utilizada em conjunto com outras metodologias (IBM, 2006).

O *OpenUp* pode ser hierarquizado em três níveis, sendo o mais alto o ciclo de vida do projeto, o mediano o ciclo de vida da iteração e o mais baixo é o micro incremento, observe a Figura 2. No micro incremento o foco é realizar pequenos pacotes de trabalhos que vão colaborar diretamente para o resultado, no ciclo de vida da iteração o objetivo é entregar um

⁴ Saiba mais sobre *steakeholders* em: <https://robsoncamargo.com.br/blog/O-que-sao-stakeholders-Saiba-tudo-sobre-eles-e-sua-importancia>. Acessado em: 01 de maio de 2021.

incremento utilizável derivado das ações do micro incremento e ciclo de vida do projeto que visa a entrega completa do produto gerado pelo projeto (IBM, 2006).

Figura 2 - Processo *OpenUp*



Fonte: (IBM, 2006).

Segundo a IBM (2006, online), o ciclo de vida do projeto para o *OpenUp* é dividido em quatro fases:

- *Inception*: essa é a primeira fase do projeto e uma característica notável nesse fase é o risco de falha do projeto muito alto e um valor baixíssimo, nessa etapa é identificado o que fazer no projeto, quais são os requisitos de maior valor; são exibidas uma possível solução para os requisitos e a estimativa do projeto.
- *Elaboration*: na segunda fase do projeto nota-se que o risco do projeto decai e o valor tem uma alta já devido a definições da primeira fase e também porque nesta etapa um dos objetivos é mitigar os riscos do projeto e, além disso, detalhar melhor os requisitos e definir o projeto de arquitetura do *software*;
- *Construction*: nesta fase o produto é construído baseado nas informações adquiridas nas fases anteriores e é importante destacar que essa fase o produto

é desenvolvido iterativamente e de forma incremental. Nesta etapa também vemos os riscos do projeto quase estáveis e a etapa que o valor mais aumenta;

- *Transition*: na última fase do projeto em que o objetivo é realizar a entrega do produto gerado e entender e documentar as lições aprendidas com o projeto realizado.

1.4 Frameworks com foco na identificação e entrega de valor de produto

A utilização de um processo híbrido para desenvolvimento de *software* em uma FSA se tornou essencial no contexto da FTT, uma vez que o *Scrum* abrange a área de processo gerencial e o *OpenUp* o processo produtivo (SOUSA et al., 2018). No entanto nenhuma das duas metodologias tem o foco na identificação ou entrega de valor do produto, pois, como afirma Pocivi (2011, p.77), “O *Scrum* e o *OpenUp* foram selecionados, principalmente, por permitirem executar, do princípio ao fim, todas as fases e as respectivas atividades do ciclo de vida de desenvolvimento de sistemas, em pequenos espaços de tempo.”.

Sendo assim, essas metodologias tem o foco na agilidade de entrega do produto, por essa razão é necessário complementar o processo quando o objetivo é a identificação e/ou entrega de valor para o cliente.

1.4.1 Design Thinking

O *Design Thinking* (DT) é um *framework* o qual tem o objetivo de criar soluções inovadoras, principalmente porque foi criado se baseando na visão que os designers teriam no momento de concepção de um novo produto, assim o processo de concepção de um novo produto para o DT deve ser realizado por uma equipe multidisciplinar para ampliar a possibilidade de novas ideias e garantindo a pluralidade de visões (VIANNA et al., 2012).

Empatia, colaboração e experimentação são os pilares do DT devendo estar presentes ao longo de todo o processo sugerido para a concepção e manutenção do produto que será desenvolvido. O pilar da empatia, considerado o mais importante, é o que permite os profissionais se colocar no lugar do cliente para compreender melhor suas necessidades e propor soluções, o pilar da colaboração é aquele que coloca em destaque a importância do trabalho em equipe e na necessidade de envolvimento de todos no processo e, por último, o pilar da experimentação que tem o objetivo de experimentar ideias e cenários diferentes para fomentar a criatividade nas soluções de problemas (UPIS, 2019).

O DT é composto por cinco etapas, que são: 1) Empatia: etapa com o foco de entender as dificuldades que determinado cliente ou usuário enfrenta; 2) Definição: nesta etapa define-se a necessidade e as motivações do usuário final; 3) Idealização: é a etapa a qual todas ideias de solução para problemas do usuário final são aceitas com a finalidade de expandir a possibilidade de ação, utiliza como ferramenta o *Brainstorm*⁵; 4) Prototipação: este é o momento de se criar algo utilizável para usuário; 5) Testar: nesta etapa são coletados os feedbacks dos usuários com a finalidade de reiniciar o ciclo e manter o produto em constante evolução (SOUZA, 2016).

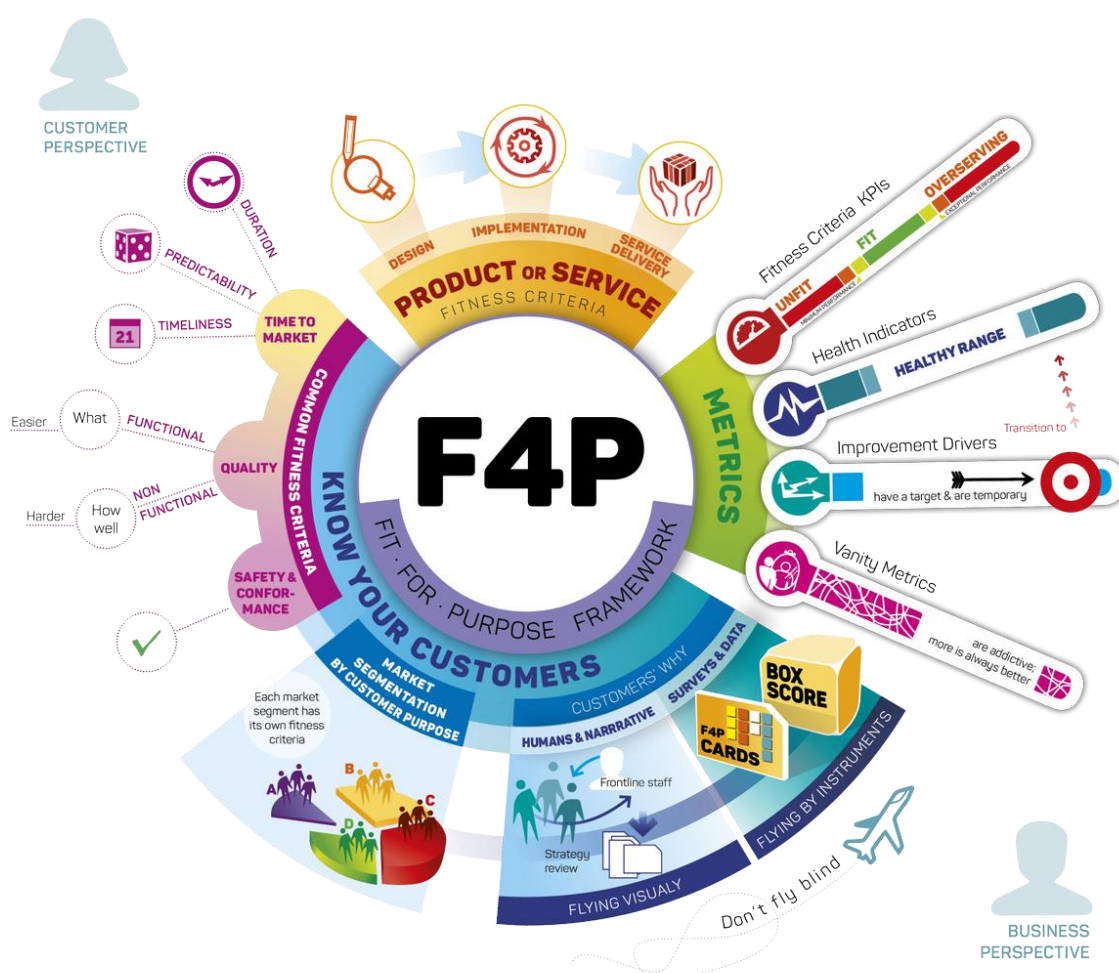
Em uma pesquisa realizada na empresa Positivo Informática por Bonini e Sbragia (2011) relata que 89% dos colaboradores concordam que as soluções geradas são mais criativas, 81% acreditam ser mais eficaz a utilização do DT no processo de criação de produtos, 78% acreditam diminuir os riscos e 74% afirmam geração de retornos financeiros. Esses números comprovam a eficácia ao implementar esse *framework* em empresas de tecnologia.

1.4.2 *Fit For Purpose*

David Anderson e Alexei Zheglov, criaram, em 2017, o *framework Fit For Purpose* (F4P) para cobrir a lacuna de eficácia dos negócios em relação a produto e orientar os negócios sobre o propósito do cliente entregando um produto que se adeque a sua necessidade (ANDERSON; ZHEGLOV, 2018). Como observado na Figura 3, o *framework* é dividido em três grandes áreas, que são o *product or service* (produto ou serviço, representado pela parte laranja), *metrics* (métricas, representado pela parte verde) e *know your customer* (conheça seu cliente, representado pela parte azul).

⁵ “Traduzida para ‘Tempestade de ideias’, tem como objetivo principal capturar muitas ideias de maneira rápida e bruta (não lapidada) e funciona muito bem em equipes” (MELO; ABELHEIRA, 2015).

Figura 3 - Infográfico *framework* F4P



Fonte: (ANDERSON; ZHEGLOV, 2018).

Para garantir que um produto ou serviço se adeque a necessidade do cliente e, como consequência, consiga manter um negócio de sucesso é necessário que os gestores desse consigam entender e balancear três momentos do processo de desenvolvimento, que são: 1) *Design (Design)*: momento com objetivo de entender quais as funcionalidades aquele produto se propõe a resolver; 2) *Implementation (Implementação)*: neste momento será possível focar em como o produto será, se aproximando ao máximo do que foi idealizado no *Design*; 3) *Service Delivery (Entrega de Serviço)*: etapa com o foco na experiência ao utilizar o produto e, também, coleta de feedbacks (DUBOVSKI, 2019).

Além de entender e conseguir dar a mesma importância ao *design*, implementação e entrega de serviço é necessário ainda garantir que o produto ou serviço esteja atingindo o *Fitness Criteria* do usuário final (DUBOVSKI, 2019). Para Anderson e Zheglov (2018) as

métricas são essenciais para entender o propósito do cliente e manter o negócio estável, logo eles dividiram as métricas em quatro categorias:

- *Fitness criteria KPIs* (métricas chaves de adequação): são as métricas que direcionam o objetivo do negócio e, normalmente, são os mesmos critérios que o consumidor usa para selecionar seu produto ou serviço. Para auxiliar nesse processo de identificação dessas métricas é aconselhável o uso de KPIs⁶;
- *Health indicators* (métricas de saúde): são as métricas essenciais para a empresa desenvolvedora, mas não tão relevantes para o usuário final, sendo assim elas garantem que o negócio está “saudável”;
- *Improvement drivers* (métricas de melhoria): são as métricas geradas a partir de alguma anomalia no negócio que gere risco para a experiência do usuário, portanto deve ser acompanhada para aquela anomalia desapareça e volte ao estado normal;
- *Vanity metrics* (métricas de vaidade): são as métricas sem relevância para nenhuma das partes envolvidas no negócio e são as que geram a sensação positiva, mas seu uso deve ser consciente, pois ela pode apresentar falsos positivos.

Com o intuito de conhecer o usuário final do produto o F4P propõe identificar se os critérios de adequação do usuários ao produto estão sendo atingidos, qual a segmentação do mercado aquele usuário pertence e se ele está dentro do público alvo e estabelecer uma relação com o usuário constante, mas com atenção em mesclar interação humana com formulários de pesquisa para não gerar uma situação de experiência negativa (DUBOVSKI, 2019).

1.5 Observações finais

A definição sobre o que é um produto de *software* e suas classificações faz-se necessário para entender sobre qual perspectiva esse trabalho tratará um produto, já o conceito de Fábrica de *Software* e os *frameworks* utilizados por ela explicita a base dos processos produtivos e gerenciais já utilizados e, por último, os *frameworks* voltados para a

⁶ Para mais informações acesse <https://rockcontent.com/br/blog/kpi/>.

identificação de valor do produto para o usuário final limitam o escopo, dada a grande quantidade de *frameworks* ágeis existentes no mercado.

Faz-se necessário para este trabalho compreender também a aplicação da disciplina ES no contexto de processo e qualidade, uma vez que o processo base definido pela ES será comparado aos processos dos *frameworks* citados para tomadas de decisão ao longo do trabalho e também compreender a importâncias das métricas na área de qualidade, pois elas serão utilizadas para o acompanhamento das alterações sugeridas. A seção seguinte apresentará a organização das atividades para alcançar estes objetivos.

3. METODOLOGIA DA PESQUISA

Tendo em vista o problema de pesquisa identificado e os objetivos estabelecidos para respondê-la, foi organizada uma estrutura de pesquisa. Na primeira etapa, foi realizada uma análise bibliográfica para identificar características do processo base definido pela Engenharia de *Software* nas metodologias *Scrum*, *OpenUp*, *Design Thinking* e *Fit for Purpose*. Essa comparação permitiu identificar pontos positivos e negativos de cada metodologia e *framework* em relação a identificação de valor de um produto para o cliente.

Em seguida foi implementado um questionário utilizando a ferramenta *google forms*, para analisar o processo de desenvolvimento de *software* da Fábrica de Tecnologias Turing (FTT). Nessa pesquisa foram levadas em consideração as informações relativas ao perfil dos colaboradores, ao nível de interação do cliente no processo de desenvolvimento, às métricas aplicadas no processo de desenvolvimento, ao objetivo geral do cliente em relação ao produto, aos diferenciais do produto gerado e a percepção dos colaboradores sobre a importância da participação do cliente no processo de desenvolvimento de *software* (APÊNDICE A – Pesquisa inicial com colaboradores da FTT).

Esse questionário é composto por perguntas objetivas e foi direcionado para colaboradores e orientadores da FTT, além disso foi criado com base nos objetivos dessa pesquisa e também em outros trabalhos desenvolvidos na FTT, como por exemplo questionário aplicado por Almeida e Batista (2020) para mapear o perfil dos colaboradores e orientadores. As respostas obtidas foram comparadas com o processo de desenvolvimento da FTT e também com o *framework* escolhido para a aplicação.

Com as respostas do questionário foi necessário levantar mais informações sobre o processo de desenvolvimento da FTT, por isso foi feita uma análise para identificar os pontos de alteração no processo de desenvolvimento de *software* da FTT, assim como métricas para acompanhamento do objetivo geral do projeto, métricas para acompanhamento da equipe, métricas de vaidade e processos de acompanhamento de feedback do cliente. Ao concluir a análise foi aplicado o *framework* escolhido em conjunto com a coordenação da FS, logo foi feito um estudo de caso sobre a aplicação de *framework*.

No intuito de apresentar o *framework*, agregar conhecimentos para os integrantes e levantar hipóteses de impedimentos que possam ser gerados foi feita uma apresentação de conteúdo, em método expositivo e com o auxílio de slides e, por último, foi aberto um

4. DESENVOLVIMENTO

A pesquisa foi estruturada para ser dividida em fases, essas fases se relacionam com cada tópico desta sessão. Sendo assim, a primeira fase desenvolvida da pesquisa foi a fase de análise comparativa dos *frameworks*, metodologias e processo base da Engenharia de *Software* (ES) apresentados pela fundamentação teórica, a segunda fase é uma análise do processo da Fábrica de Tecnologias Turing (FTT) para identificar pontos estratégicos para aplicação da pesquisa e a terceira a aplicação das alterações propostas e análise dos resultados coletados durante a aplicação.

3.1 Análise comparativa dos *frameworks*

A primeira etapa de pesquisa é comparar os processos das metodologias e dos *frameworks* apresentados na fundamentação teórica com a finalidade de identificar se seguem o processo mínimo de desenvolvimento de *software*, identificar características que auxiliam na descoberta de valor do produto e selecionar o *framework* para ser aplicado no processo de desenvolvimento da FTT.

Para Sommerville (2011), as etapas de especificação de requisitos, projeto e implementação de *software*, validação de *software* e evolução de *software* devem estar presentes em todo processo de desenvolvimento de *software*. Baseado nesta afirmação e na necessidade de entender o cenário do problema levantado para essa pesquisa foi criada a Figura 4 que representa uma comparação entre as etapas do processo base definido pela Engenharia de *Software* (ES) com as etapas dos *frameworks* *Design Thinking* (DT) e *Fit for Purpose* (F4P) e das metodologias *Scrum* e *OpenUp*.

Figura 4 - Comparação do processo base de ES com os processos dos *frameworks* e metodologias

Processo base definido pela Engenharia de Software			
Especificação de Requisitos	Projeto e Implementação de software	Validação de software	Evolução de software
Planejamento geral e projeto de arquitetura			
Ciclo iterativo e incremental (avaliar, selecionar, desenvolver e revisar)			Entrega do projeto
Inception			
	Elaboration	Construction	Transition
Empatia	Definição	Idealização	Prototipação
			Testar
Design	Implementation	Service Delivery	
Legenda: ■ Scrum ■ OpenUP ■ Design Thinking ■ Fir for Purpose			

Fonte: Autoria própria.

A etapa de especificação de requisitos definida pelo processo base de ES se relaciona diretamente com etapas de planejamento geral e projeto de arquitetura (*Scrum*), ciclo iterativo (*Scrum* – avaliar), *inception* (*OpenUp*), *elaboration* (*OpenUp*), empatia (DT), definição (DT) e *design* (F4P). Essa relação existe, pois todas essas fases citadas são direcionadas a gerar um artefato que detalhe aspectos do produto que será desenvolvido, sendo assim esses artefatos podem ser compreendidos como uma especificação de requisitos, já que para Pressman (2011) uma especificação de requisitos do sistema “pode ser um documento por escrito, um conjunto de modelos gráficos, um modelo matemático formal, um conjunto de cenários de uso, um protótipo ou qualquer combinação dos fatores citados.”.

Para Sommerville (2011), a etapa de projeto e implementação de *software* que é a fase “em que o *software* é projetado e programado”, logo esta fase pode ser comparada com as fases de ciclo iterativo (*Scrum* – selecionar e desenvolver), *elaboration* (*OpenUp*), *construction* (*OpenUp*), idealização (DT), prototipação (DT) e *implementation* (F4P), sendo todas essas fases tem o foco em entregar um incremento funcional.

A etapa de validação de *software* tem o objetivo de validar se o produto gerado pela etapa de planejamento e implementação de *software* atende o que foi definido na etapa de especificação de requisitos e as etapas das metodologias e *frameworks* que correspondem a essa definição são: ciclo iterativo (*Scrum* – revisar), *construction* (*OpenUp*), testar (DT) e *service delivery* (F4P).

A última etapa para o desenvolvimento de *software* definida para ES é a etapa de evolução do *software* que tem o objetivo de colher *feedbacks* e apontar pontos de melhoria, dessa forma as etapas das metodologias e *frameworks* que correspondem a essa definição são: entrega de projeto (*Scrum*), *transition* (*OpenUp*), testar (DT) e *service delivery* (F4P).

Em comparação dos *frameworks* podemos perceber que todos eles percorrem todas as etapas mínimas, defendidas pela Engenharia de *Software*, para o desenvolvimento de um determinado produto. No entanto, o que leva a selecionar o *framework Fit for Purpose* para aplicação no processo de desenvolvimento da FTT é foco na entrega de valor para o cliente, a necessidade de metrificação do processo e a definição de valor do produto, que para o F4P não deve somente focar no produto gerado, mas também na sua produção e toda forma de interação com o cliente.

3.2 Análise do processo de desenvolvimento da FTT

O processo de desenvolvimento da Fábrica de Tecnologias Turing (FTT) é um processo híbrido baseado em *Scrum* e Open UP, sendo que, para Pocivi (2011), “*Scrum* é fundamentado na teoria de controle de processos empíricos, enfatizando o gerenciamento de projetos” (p. 28) e o *Open Up* uma metodologia focada no processo produtivo em que a participação e envolvimento dos *stakeholders* é essencial para o sucesso do projeto.

A Figura 5 apresenta o processo de desenvolvimento de *software* da FTT, este processo “foi estruturado visando explorar as melhores práticas de cada uma das metodologias que o compõe, criando uma *framework* iterativo-incremental e adaptativo para o desenvolvimento de *software* por acadêmicos em todos os estágios do conhecimento” (ALMEIDA; BATISTA, 2020).

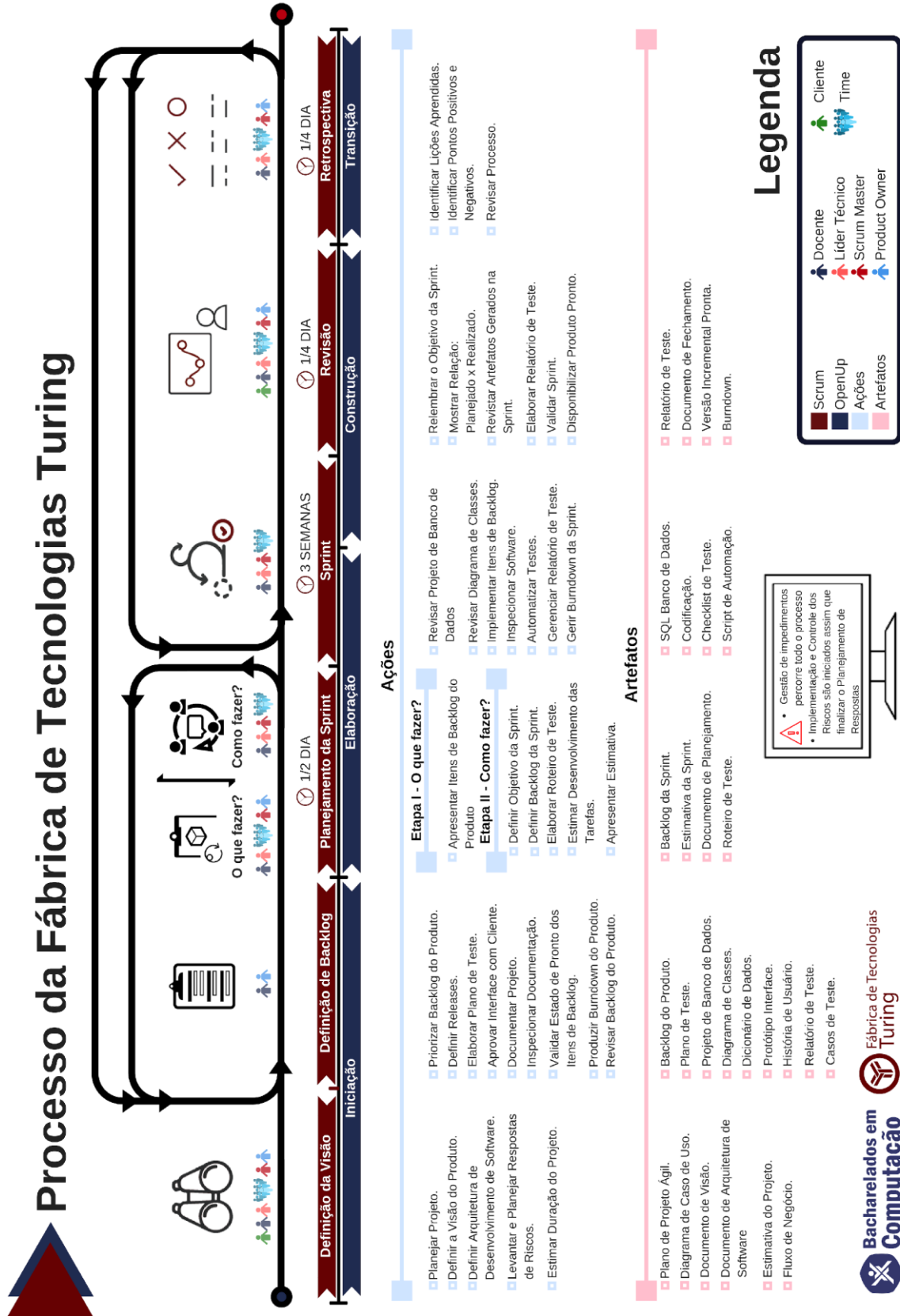
Sendo assim esse processo é baseado em dois *frameworks*, *Scrum* (representado pela cor vinho na figura 5) e Open Up (representado pela cor azul escuro na figura 5). Nesse momento a análise do processo de desenvolvimento da FTT tem o foco de identificar onde o cliente deveria participar do processo e como o processo é metrificado.

Segundo o processo de desenvolvimento da FTT os clientes devem participar na definição de visão do produto e também na revisão. A participação do cliente na definição da visão do produto é importante, uma vez que a nesse momento é definido a razão do produto existir se baseando na visão e experiência do *Product Owner* (P.O) e na necessidade do usuário enquanto cliente (TORRES, 2016).

Outro passo do processo que o cliente deve participar é na revisão de produto, pois nesse momento o *Scrum team* é responsável por apresentar todo o resultado da *sprint* para os principais envolvidos no projeto, sendo assim também é definido sobre as futuras implementações e qual o progresso em relação a meta do produto (SCHWABER; SUTHERLAND, 2020), por essa razão é interessante o cliente participar e opinar sobre o resultado e futuras implementações.

A Figura 5 detalha os passos do processo de desenvolvimento de *software* utilizado pela FTT, mas não deixa explícito informações importantes do processo como a metrificação ou acompanhamento de *feedback* do cliente. Por essa razão foi necessário aplicar uma pesquisa para mapear o perfil dos colaboradores, participação dos clientes no processo, percepção dos colaboradores sobre a importância da participação do cliente, métricas envolvidas no processo e identificar se havia um subprocesso de acompanhamento de *feedback* dos clientes (APÊNDICE A – Pesquisa inicial com colaboradores da FTT).

Nessa pesquisa as métricas foram divididas em métricas de objetivo geral, métricas de performance da equipe e métricas de vaidade, no entanto os orientadores da FTT afirmam que não há nenhum tipo de métricas no processo de desenvolvimento. Além disso, nessa mesma pesquisa, os orientadores ainda afirmam que não há um processo de análise e acompanhamento do *feedback* do cliente.



Fonte: (FARIA et al., 2018).

3.3 Proposta e aplicação das alterações propostas no processo de desenvolvimento

Como a análise do processo desenvolvimento da FTT e a pesquisa com os colaboradores apontaram a ausência da metrificação do processo e ausência de um subprocesso para acompanhamento de *feedback* do cliente, que é de extrema importância para o *framework* que será aplicado (*Fit for Purpose* – F4P), foi proposto uma implementação dividida em duas frentes. A primeira frente é relativa aplicação de métricas no processo para acompanhar a evolução do produto e a segunda frente voltada aplicar um subprocesso análise e acompanhamento do *feedback* dos clientes.

Antes de fazer a aplicação das alterações propostas fez-se necessário selecionar qual projeto aplicar as mudanças e, no momento da pesquisa, haviam dois projetos em andamento no momento da pesquisa. O primeiro projeto é o Sites Colégios, com foco em desenvolver sites dinâmicos para outras instituições da Associação Educativa Evangélica (AEE), e o segundo é o Guardas, projeto voltado para desenvolver uma aplicação para gestão da equipe de segurança que atua na Universidade Evangélica de Goiás (uma das instituições pertencentes à AEE).

O projeto Guardas não foi possível utilizar dado que ele estava sendo utilizado em outra pesquisa de graduação, logo para que os dois trabalhos não sofressem influências foi necessário utilizar o projeto Sites Colégios. Após a definição do projeto, foi apresentado para todos os colaboradores sobre o F4P, sobre as alterações propostas e sobre como as mudanças iriam impactar no cotidiano deles.

No entanto, quando a proposta das alterações ainda estava sendo elaborada o projeto Sites Colégios sofreu um impedimento de arquitetura que bloqueou o andamento do projeto como um todo impedindo de continuar o seu desenvolvimento e o projeto Guardas já tinha sido utilizado para a outra pesquisa e não sofreria mais impacto. Para concluir minha pesquisa foi necessário mudar de projeto e aplicar as alterações propostas no projeto Guardas, sendo assim essa aplicação ainda ficou dividida nas duas frentes citadas anteriormente e elas serão descritas a seguir.

3.3.1 Primeira frente – Metrificação do processo de desenvolvimento da FTT

Para o *framework Fit for Purpose* (F4P) as métricas são divididas em métricas chaves de adequação, métricas de saúde, métricas de melhoria e métricas de vaidade (ANDERSON; ZHEGLOV, 2018). Por essa razão, foi necessário definir pelo menos 4 métricas para aplicar

no processo de desenvolvimento da Fábrica de Tecnologias Turig (FTT), para analisar se o processo estava atendendo ou não as expectativas durante o desenvolvimento do *software*, cada uma dessas métricas será apresentada a seguir com o justificativa sobre o porquê delas serem escolhidas.

Toda empresa deve ter uma equipe alinhada e direcionada para atingir os objetivos, sendo assim um *framework* que define esses objetivos de maneira ágil e com a participação de todos os colaboradores é o *Objective and Key Results* (OKR). Como o OKR define as metas gerais de uma empresa e/ou projeto é necessário definir os *Key Performance Indicator* (KPI) de cada OKR definido, logo os KPI se tornam métricas para serem acompanhadas ao longo do projeto e entender se o objetivo geral está sendo cumprido (COUTINHO, 2021).

Para aplicação nessa pesquisa foi disponibilizado via ferramenta de apresentações do google um cartão para a equipe definir os OKR e KPI do projeto em que será aplicada a pesquisa, logo o cartão de OKR (Figura 6) era composto por um objetivo geral (OKR) e até três métricas (KPI) para cada objetivo geral para serem acompanhadas ao longo do projeto.

Figura 6 - Cartão de OKR

Objective and Key Results - OKR

Key Performance Indicator- KPI

- 1 - _____
- 2 - _____
- 3 - _____

Fonte: autoria própria.

Métricas de saúde são as métricas que são necessárias para garantir a eficiência e eficácia da equipe no processo de desenvolvimento, sendo assim são métricas que são de extrema importância para o projeto, mas que o foco é na performance da própria equipe e não no negócio como um todo (ANDERSON; ZHEGLOV, 2018). Uma métrica de melhoria é

uma métrica de saúde que por alguma consequência está fora dos parâmetros estabelecidos, logo deve ser trabalhada para retornar a classe de métrica de saúde (DUBOVSKI, 2019).

Logo para essas duas classes foi necessário definir mais duas métricas, então foi proposto implementar *Work in Progress* (WIP) para toda a equipe envolvida e *Lead Time*, inicialmente, da fase de codificação do *software*. Essas duas métricas tem a finalidade de limitar o trabalho e tempo de entrega das atividades, garantindo assim a cadência de entrega e diminuindo as incertezas em relação ao projeto, pois permitem ter uma visão mais focada no que realmente precisa ser entregue (ALBINO, 2018).

A equipe do projeto Guardas tem 3 desenvolvedores, 2 analistas de qualidade, 2 analistas de requisitos, 1 *Product Owner*, 1 *Scrum Master* e 1 analista de banco de dados. Com essas informações, inicialmente as atividades devem ser consideradas para WIP quando são priorizadas no *sprint backlog* até a fase de revisão da *sprint* com a meta 6 atividades por *sprint*, posteriormente essa meta pode ser alterada de acordo com a necessidade do projeto, necessidade do cliente e/ou tamanho de equipe. Para o *lead time*, inicialmente, ficou definido que o *lead time* máximo para a etapa de desenvolvimento de uma atividade é o período da *sprint*, 10 dias úteis.

Já as métricas de vaidade para Dias (2016) “poderiam ser descritas como toda informação coletada e acompanhada por profissionais, que na verdade não ajudam a tomar decisão nenhuma”, logo essas métricas não podem ser ignoradas, mas analisá-las com tanto foco pode não trazer o retorno esperado para o negócio. Por essa razão, para conseguir exemplificar a aplicação das métricas de vaidade foi decidido medir a quantidade de *commits* que a equipe de desenvolvimento realiza durante a *sprint*.

No entanto, o WIP, *Lead Time* e a quantidade de *commits* na *sprint* não foram possíveis implementar, uma vez que para gerar resultados dessas métricas era necessário que fossem analisadas pelo menos duas *sprints* do projeto para observar a evolução das métricas e no contexto dessa pesquisa, onde houveram impedimentos nos projetos que ocasionaram o atraso na pesquisa, foi possível implementar somente as métricas de chave de adequação.

3.3.2 Segunda frente – Subprocesso para análise e acompanhamento de feedback do cliente

Para um produto ter valor no mercado e atender às necessidades dos clientes é necessário que haja um processo para analisar a viabilidade do desenvolvimento daquela alteração e também acompanhar os *feedbacks* dos usuários para que o negócio não seja

direcionado por achismos (ANDERSON; ZHEGLOV, 2018; RODRIGUES, 2014). Para conseguir aplicar esse subprocesso no processo de desenvolvimento da FTT, foi decidido que seria utilizado o *Fit Box Score* (FBS).

O FBS é um tipo de pesquisa baseado no *Net Promote Score*⁷ (NPS), uma métrica de lealdade do cliente com foco em entender sua satisfação em relação ao produto/serviço oferecido, logo o FBS foi idealizado para ser um cartão simples em que o cliente não tenha dificuldades em entender e preencher, nem tanto se sinta entediado respondendo às perguntas e, apesar de ser um cartão, isso não limita a FBS ser executado somente pelo cartão, ele pode ser executado numa simples conversa com o cliente (ANDERSON; ZHEGLOV, 2018; SILVA, 2020).

O cartão do FBS é composto por três partes, em que a primeira tem o objetivo de identificar pelo menos três motivações do porquê o cliente escolheu o produto, a segunda parte o foco é entender o quanto ele acha que as razões pela qual ele buscou aquele produto foram atendidas e, por último, a terceira parte busca o motivo pelo qual ele classificou o produto como bom ou ruim (TOLMASQUIM, 2019). A Figura 7 apresenta o modelo do cartão disponibilizado para a aplicação do FBS.

Figura 7 - Cartão *Fit Box Score*

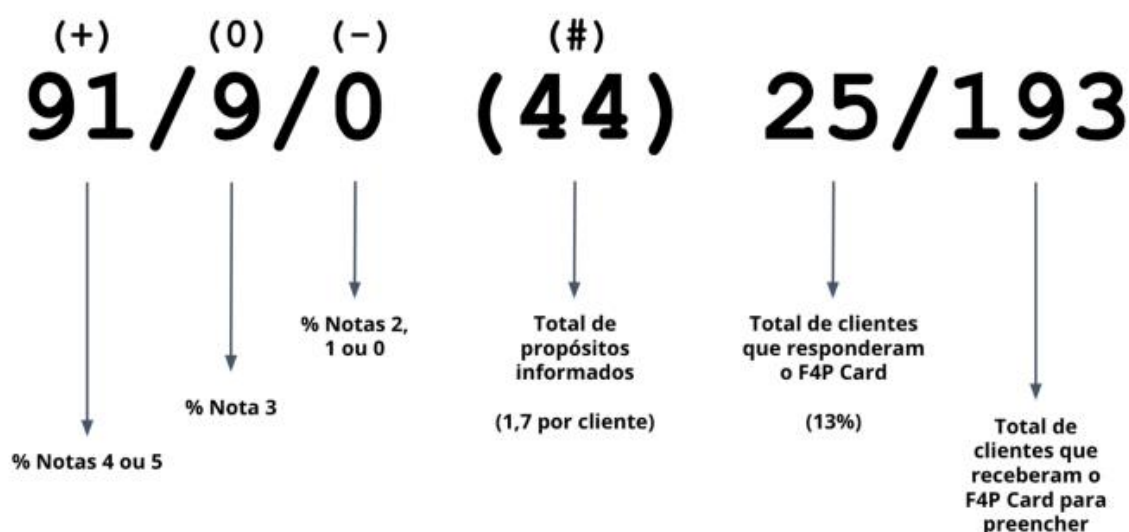
Escolha até três razões para você ter escolhido nossos serviço/produto.		1 - _____
		2 - _____
		3 - _____
Fit Box Score	Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1 2 3
	Foram superadas	
	Foram totalmente atendidas	
	De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores	
	Algumas necessidades importantes não foram atendidas	
	A maior parte das minhas necessidades não foram atendidas	
Nada foi útil		
Qual foi o motivo para você escolher cada nota?		1 - _____
		2 - _____
		3 - _____

Fonte: autoria própria.

⁷ Para entender mais sobre NPS acesse <https://resultadosdigitais.com.br/blog/o-que-e-nps/>.

Com o cartão respondido os resultados precisam ser analisados, sendo assim para o FBS os resultados se tornam métricas que devem ser acompanhados a cada iteração ou como uma rotina de coleta de *feedback*, logo para essa pesquisa foi definido que seria entregue essas cartões para o cliente responder toda revisão de *sprint*. A análise de resultados é baseada no modelo de pontuação de baseball americano⁸ (Figura 8). Logo, a leitura da figura pode ser feita da seguinte forma: De 193 cartões distribuídos 25 foram respondidos, nesses 25 cartões respondidos foram citados 44 propósitos em que 91% dizem que o produto atendeu ou superou a expectativa do cliente, 9% tiveram a experiência mediana e 0% teve uma péssima experiência ou não tiveram seus objetivos atingidos (DUBOVSKI, 2019).

Figura 8 - Exemplo *box score*



Fonte: (DUBOVSKI, 2019).

Com a aplicação da primeira frente será possível acompanhar se o processo de desenvolvimento está ocorrendo de maneira fluida, com entregas de incrementos dentro do prazo estabelecido, alinhados com o objetivo da empresa e cliente e com a qualidade necessária e com a aplicação da segunda frente será possível entender a necessidade do cliente e acompanhar a melhoria do produto de acordo com sua necessidade.

⁸ Esse modelo de pontuação tem o objetivo de detalhar o resultado, trazendo mais informações do que o próprio resultado final (ANDERSON; ZHEGLOV, 2018).

5. RESULTADOS E DISCUSSÕES

No primeiro momento foi feita uma análise bibliográfica com os *frameworks Scrum, OpenUp, Design Thinking (DT) e Fit for Purpose (F4P)* comparando com o processo base da Engenharia de *Software (ES)*, nessa comparação foi possível observar que todas as metodologias analisadas percorrem o processo mínimo exigido pela ES. No entanto, ao comparar as metodologias com a definição de valor do produto utilizada para esse trabalho, foi possível identificar que somente o F4P evidencia que o valor do produto não é unicamente o produto gerado e sim toda a experiência que o cliente tem com a empresa, desde o entendimento da necessidade do cliente até a utilização do produto, assim como defende Ries (2012).

A deficiência na identificação de valor do produto fica mais evidente quando, ao analisar o processo de desenvolvimento da Fábrica de Tecnologias Turing (FTT), é notado que a participação do cliente é exigida somente na reunião de planejamento do projeto e nas reuniões de revisão da *sprint*, além disso o processo de desenvolvimento da FTT não define nenhum tipo de metrificação para garantia da “saúde” do processo e nem possui um subprocesso claro de coleta e análise de *feedback* dos clientes.

Esses pontos levantados contradizem a frase o lema do F4P, “*Don’t fly blind*” (Não voe às cegas – Tradução livre) de Anderson e Zheglov (2018), criadores do *framework* F4P, em que eles defendem que o processo de desenvolvimento é tão importante quanto entender a necessidade do cliente, pois ele também afeta a experiência final do cliente e o processo deve ser constantemente acompanhado. Para Peralta (2020) ter o processo de desenvolvimento alinhado com a necessidade do cliente é uma das maiores dificuldades no desenvolvimento de *software* atualmente.

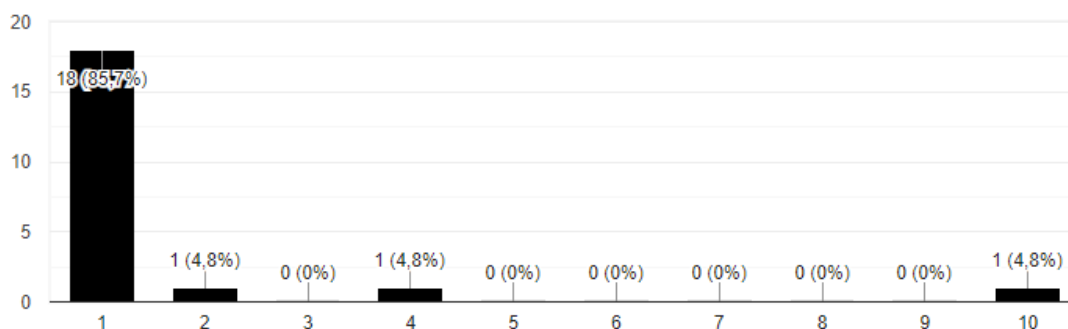
A aplicação da pesquisa de ambientação com os colaboradores e orientadores da FTT possibilitou identificar, que apesar do processo exigir a participação dos clientes em determinadas etapas do processo de desenvolvimento, na realidade o cliente não participa ou quase não participa de nenhuma etapa do processo. A realidade apresentada contraria todas as metodologias, *frameworks* e processo base da engenharia de *software* estudados para esta pesquisa, uma vez que cada uma delas possui uma fase específica para coletar e analisar os *feedbacks* dos clientes (ANDERSON; ZHEGLOV, 2018; SCHWABER; SUTHERLAND, 2020; SOMMERVILLE, 2011).

Na pesquisa foram analisados, na visão dos colaboradores, se o cliente tinha participação nas reuniões de planejamento do projeto e/ou sprint, reuniões de revisão do produto e reuniões diárias. Nas reuniões de planejamento de projeto e/ou sprint (Gráfico 1) 85,7% dos colaboradores afirmam que o cliente não participa de forma alguma dessas reuniões, 4,8% afirmam quase nunca participam, 4,8% afirmam que participam pouco e 4,8% afirmam que os clientes participam de todas as reuniões de planejamento propostas.

Gráfico 1 - Participação dos clientes nas reuniões de planejamento

O cliente participa das reuniões planejamento (sprint ou do projeto)?

21 respostas



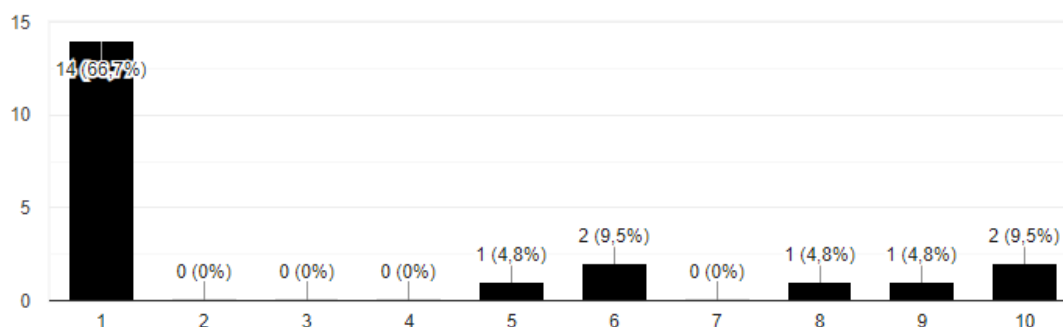
Fonte: Autoria própria.

Já nas reuniões de revisão o cenário é um pouco mais positivo (Gráfico 2), mas ainda não deixa de ser preocupante, pois 66,7% dos colaboradores que responderam à pesquisa afirmam que o cliente não participa de forma alguma dessas reuniões, 4,8% afirmam que os clientes participam de maneira mediana, 9,5% dizem que participam as vezes, 4,8% afirmam que os clientes são participativos nessas reuniões, 4,8% dizem que os clientes participam de quase todas as reuniões de revisão e 9,5% afirmam que os clientes estão presentes em todas as reuniões propostas.

Gráfico 2 - Participação dos clientes nas reuniões de revisão

O cliente participa das reuniões de revisão do produto?

21 respostas



Fonte: Autoria própria.

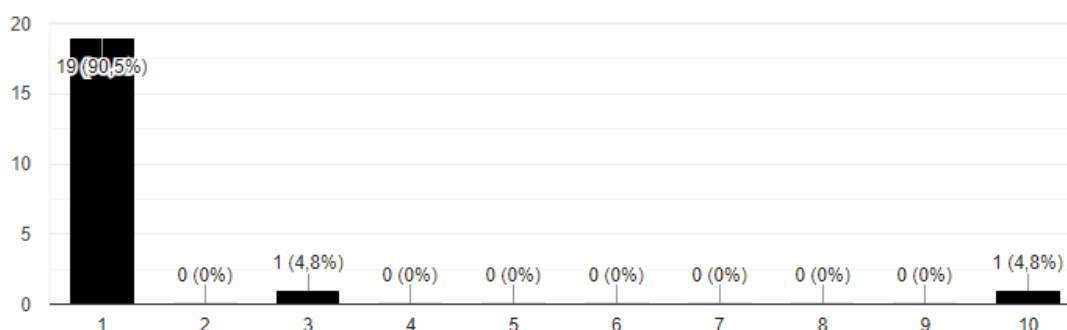
Nas reuniões diárias a participação do cliente é opcional, mas faz necessária em momentos oportunos para acompanhar fases críticas do projeto (Gráfico 3), nota-se que para 95,5% dos colaboradores os clientes nunca participaram de nenhuma reunião diário, 4,8% acreditam que o cliente participam muito pouco e 4,8% dizem que os clientes participam de todas as reuniões propostas.

Gráfico 3 - Participação dos clientes nas reuniões diárias

O cliente participa das reuniões diárias?



21 respostas



Fonte: Autoria própria.

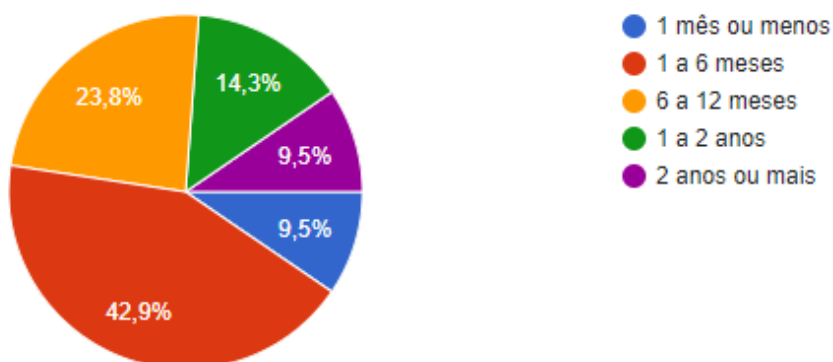
Ao todo foram 21 colaboradores da FTT que responderam essa pesquisa, logo há alguns fatores que podem ter influenciado nos resultados anteriores assim como o tempo de experiência no ambiente da Fábrica de Tecnologias Turing, tempo de experiência em outras empresas e conhecimento sobre o processo de desenvolvimento. Logo de 100% (21) colaboradores que responderam à pesquisa 9,5% (2) trabalham na FTT a menos de um mês,

42,9% (9) trabalham de 1 a 6 meses, 23,8% (5) trabalham de 6 a 12 meses, 14,3% (3) trabalham de 1 a 2 anos e 9,5% (2) trabalham a mais de 2 anos na FTT (Gráfico 4).

Gráfico 4 - Experiência dos colaboradores dentro da FTT

Há quanto tempo trabalha na FTT?

21 respostas



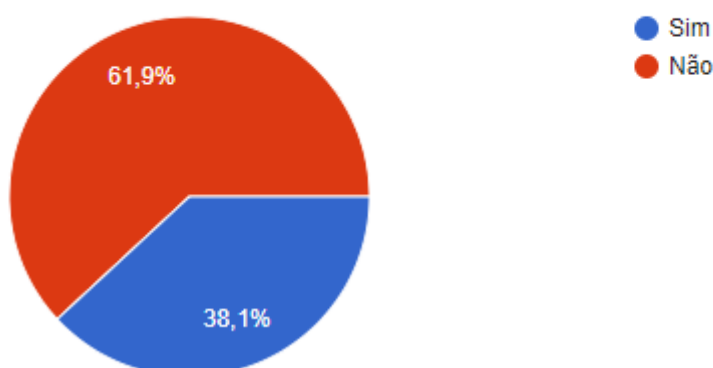
Fonte: Autoria própria.

Em relação a experiências dos colaboradores fora do ambiente da FTT foi observado (Gráfico 5) que 61,9% (13) nunca tiveram nenhuma experiência fora da FTT e dos 38,1% (8) que afirmam ter experiência em outros ambientes possuem, em média, 9 meses trabalhando fora do ambiente da FTT.

Gráfico 5 - Quantidade de colaboradores que já trabalharam em outros ambientes

Possui experiência em outro ambiente de desenvolvimento fora da FTT?

21 respostas



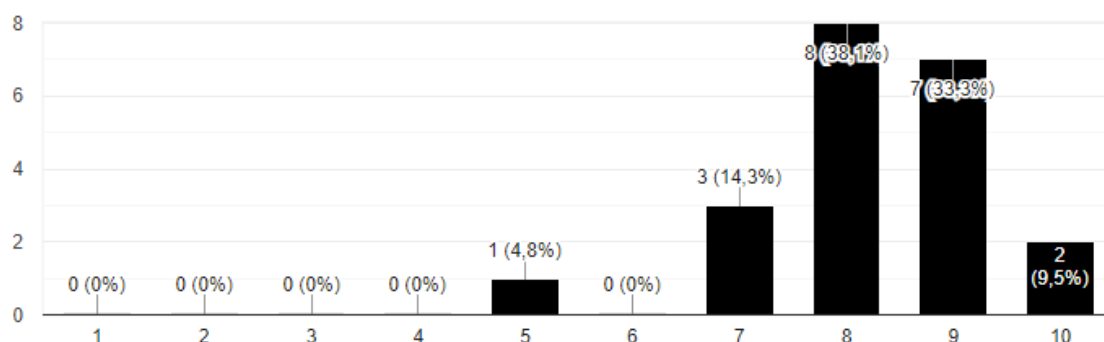
Fonte: Autoria própria.

Ao analisar o cenário sobre o conhecimento dos colaboradores em relação ao processo o cenário é um pouco mais otimista, pois 4,8% afirmam conhecer o processo de maneira mediana, 14,3% afirmam conhecer o processo bem sem muita profundidade, 38,1% afirmam conhecer o processo com pouco mais de profundidade, 33,3% afirmam entender o processo completamente e 9,5% afirmam dominar o processo como um todo sendo capazes de propor melhorias para ele (Gráfico 6).

Gráfico 6 - Nível de conhecimento dos colaboradores sobre processo de desenvolvimento da FTT

Na sua opinião, quanto você conhece do processo de desenvolvimento utilizado pela FTT?

21 respostas



Fonte: Autoria própria.

Logo, com o tempo de experiência dentro da FTT, que é em média 10 meses, o tempo de experiência fora da FTT, que é em média 9 meses, e o com o que o colaboradores dizem conhecer do processo é possível entender que a grande maioria deles possuem experiência profissional o suficiente para conhecer os conceitos básicos das metodologias ágeis e conseguem aplicar na FTT esses conceitos, mas ainda há aqueles que precisam conhecer mais dos conceitos e do próprio processo de desenvolvimento da Fábrica de Tecnologias.

Além disso é necessário entender se, na Fábrica de Tecnologias Turing, os colaboradores entendem a necessidade da participação do cliente no processo de desenvolvimento, com isso na pesquisa aplicada 90,5% concordam em dizer que a participação do cliente do processo de desenvolvimento é importante e 9,5% discordam dessa afirmação, por outro lado 100% dos colaboradores entender ser importante ter uma coleta de feedbacks do cliente.

Na primeira frente aplicada foi possível definir um Objective Key and Results (OKR) com três Key Performance Indicators (KPI) representado pela Figura 9, em que os colaboradores definiram o OKR como uma métrica de performance ou melhoria, focando assim em satisfazer metas internas na FTT e não metas do produto com uma visão externa e/ou do cliente.

No entanto o intuito da pesquisa era desenvolver esses OKR com foco no produto e não na FTT como um todo, pois para Anderson e Zheglov (2018) os KPI são as métricas que conectam o mercado com o negócio, logo elas servem de direcionamento para entender quais as necessidades do cliente ainda não foram atendidas para que sejam trabalhadas e mantenha o produto em constante evolução. Com esse resultado é perceptível a cultura dos colaboradores de sempre medirem o próprio ambiente e não conseguirem analisar as necessidades do usuário.

Figura 9 - Cartão de OKR preenchido

Objective and Key Results - OKR

Obj - Ser reconhecida como uma empresa referência na consolidação de profissionais para o mercado de trabalho.

KR¹ - Ter 80% dos alunos egressos inseridos no mercado de trabalho em 2021/2.

KR² - Estruturar ao menos duas parcerias com empresas locais em 2021/2.

KR³ - Ter uma retenção dos alunos egressos de 2021/2 de pelo menos 3 meses.

Key Performance Indicator- KPI

1 - Reduzir em 60% o número de alunos desistentes.

2 - Aumentar em 100% o número de alunos participantes da capacitação em relação ao semestre anterior.

3 - Elevar a nota de satisfação da empresa para 8 pontos numa escala de 0 a 10.

Fonte: Autoria própria.

Ao aplicar o *Fit Box Score* (FBS) oito *stakeholders* do projeto foi possível obter o resultado expresso pela figura 10, logo das 8 pessoas, que eram esperadas para responder o questionário do FBS, 5 responderam gerando 10 propósitos pelos quais elas acreditavam ver no produto e que 100% das pessoas são promotoras do produto gerado pela FTT atualmente. Dentre os propósitos que se destacaram com nota máxima (5) são: A fácil e rápida forma de documentação das ocorrências pelo aplicativo, simplicidade ao utilizar o aplicativo, aplicativo intuitivo, agilidade para obter informações e facilidade na gestão e compreensão das ocorrências (Apêndice).

Figura 10 - Resultado final FBS**100/0/0 (10) 5/8**

Fonte: Autoria própria.

A aplicação do FBS ocorreu na fase de prototipação do produto, isso significa que o produto ainda não foi desenvolvido e que essa aplicação serviu para validar a ideia de produto que a equipe desenhou e se essa ideia atenderia às expectativas dos clientes finais. Isso proporcionou a equipe validar iniciar a fase de documentação e codificação sem que haja grandes mudanças ao decorrer do projeto, pois, para Gomes (2017), essas mudanças podem causar atrasos nas entregas e custos financeiros a empresa. No entanto é necessário que esses resultados sejam acompanhados quando o produto estiver desenvolvido e que seja criada uma rotina para esse acompanhamento.

6. CONCLUSÃO E CONSIDERAÇÕES FINAIS

A pesquisa possibilitou comprovar a importância de um processo para identificação de valor do produto que não fosse restrito ao produto pronto, mas que fosse relacionado a toda experiência do cliente desde a solicitação até a entrega do produto. Além disso, foi possível concluir que mesmo com um processo estruturado para identificação de valor do produto é preciso que haja uma cultura em que todos os envolvidos trabalham para agregar valor ao produto, caso contrário a visão dos envolvidos será sempre focada na própria empresa como aconteceu nas definições das métricas de OKR dessa pesquisa.

Com o estudo do processo de desenvolvimento da Fábrica de Tecnologias Turing, estudo comparativo entre os *frameworks* e com a pesquisa de ambientação aplicada com os colaboradores da FTT foi possível notar que o problema de identificação de valor do produto geralmente se dá pela não participação do cliente durante o processo de desenvolvimento como explicitado nos gráficos 1, 2 e 3.

Para mitigar esse risco é necessário que haja a metrificação do processo de desenvolvimento para que seja possível mensurar se o objetivo está sendo atingido e se o processo é efetivo ao realizar as entregas, além disso faz necessário um subprocesso para acompanhamento da opinião do cliente sobre o produto e sobre sua experiência. Sendo assim um *framework* que abrange tanto a metrificação quanto o subprocesso é o *Fit for Purpose*, por isso ele foi selecionado como base para aplicação dessa pesquisa.

Cada ambiente de desenvolvimento de *software* possui seu próprio processo e cada processo pode ser baseado em diversas metodologias, por essa razão essa pesquisa se trata de uma pesquisa específica para o ambiente da FTT e caso desejem aplicar em outro ambiente é necessário que sejam executadas cada etapa dessa pesquisa novamente, no entanto aqueles processos que forem baseados na metodologia *Scrum*, *OpenUp*, *Design Thinking* ou *Fit for Purpose* sofrem menos impacto pois eles já foram estudados nesta pesquisa.

O análise comparativa entre os *frameworks* foi importante para a pesquisa, pois com ela foi possível identificar qual a semelhança entre os *frameworks* e o processo base da engenharia de *software* defendido por Sommerville (2011), além disso permitindo identificar quais etapas eram focadas em obter a opinião do cliente para que possa realizar o acompanhamento da maneira correta.

Ao analisar o processo de desenvolvimento da FTT e, em conjunto, aplicar a pesquisa de ambientação foi possível identificar características no processo de desenvolvimento que

não condiziam com os *frameworks* estudados, que tinham o foco na geração de valor para produto. Com isso, notou-se que o processo da FTT não possuía nenhum tipo de metrificação e nenhum processo de coleta de *feedback* dos clientes o que gerou um grande risco para o projeto.

Com a aplicação das propostas dessa pesquisa foi possível metrificar e aplicar o subprocesso para acompanhamento dos *feedbacks*, mesmo que de forma embrionária. Logo, para garantir que identificação e geração de valor do produto seja efetiva é necessário que essa pesquisa seja aplicada e que seus resultados sejam acompanhados por um longo período para que assim possibilite observar a evolução do processo e das entregas de valor do produto, podendo propor novas métricas de acompanhamento.

A visão expressa por essa pesquisa deixa claro a mentalidade dos colaboradores de observar a própria empresa ou o próprio negócio, mas não conseguem observar as opiniões e objetivos impostos pelo mercado e/ou clientes. Nesse sentido, seria interessante uma pesquisa para analisar o comportamento e visão das pessoas diretamente envolvidas no desenvolvimento de *software* comparando a FTT e outros ambientes sobre a ótica de gerar valor para o produto.

A aplicação da pesquisa, desde estudos teóricos até a aplicação propriamente dita, permitiu destacar a importância de um processo voltado para a identificação de valor do produto em que o valor não está contido somente do produto gerado e sim em todo processo de desenvolvimento e experiência do cliente com a empresa.

7. REFERÊNCIAS

ALBINO, R. D. **Métricas ágeis: obtenha melhores resultados em sua equipe**. 1. ed. São Paulo, SP: Casa do Código, 2018.

ALMEIDA, P. H. P.; BATISTA, R. A. **PROCESSO DE GESTÃO DO CONHECIMENTO PARA AMBIENTE ACADÊMICO DE DESENVOLVIMENTO DE SOFTWARE**. Anápolis: [s.n.].

ANDERSON, D. J.; ZHEGLOV, A. **Fit For Purpose: How Modern Business Find, Satisfy & Keep Customers**. 2. ed. Seattle, WA: Blue Hole Press, 2018.

BONINI, L. A.; SBRAGIA, R. O Modelo de Design Thinking como Indutor da Inovação nas Empresas: Um Estudo Empírico. **Revista de Gestão de Projetos**, jun. 2011.

BORGES, K. S.; MACHADO, R. P.; CARVALHO, T. P. DE. **Programa de Extensão “Fábrica de Software Acadêmica”: contribuindo para a formação profissional na área de informática**. Porto Alegre: IFRS, 2011.

CAMARGO, R. **Metodologia Agile: quais as mais utilizadas?** Disponível em: <<https://robsoncamargo.com.br/blog/Metodologia-Agile-quais-as-mais-utilizadas>>. Acesso em: 15 nov. 2021.

COUTINHO, B. **Qual é a diferença entre OKR e KPI? Qual utilizar?** Disponível em: <<https://www.cursospm3.com.br/blog/qual-e-diferenca-entre-okr-e-kpi/>>. Acesso em: 1 nov. 2021.

DIAS, T. **Por que nós precisamos falar sobre as Métricas de Vaidade?** Disponível em: <<https://rockcontent.com/br/blog/metricas-de- vaidade/>>. Acesso em: 2 nov. 2021.

DUBOVSKI, M. **Fit for Purpose e a importância de conhecer o propósito do cliente**. Disponível em: <<https://medium.com/@dubovski/fit-for-purpose-e-a-importancia-de-conhecer-o-propósito-do-cliente-abdfd29b9baf>>. Acesso em: 2 maio. 2021.

EPPINGER, S. D.; ULRICH, K. T. **Product Design and Development**. 6. ed. Nova York, NY: McGraw-Hill Education, 2016.

FARIA, A. DE A. et al. **Guia Prático de Funcionamento da Fábrica de Tecnologias Turing (FTT)**. Anápolis: [s.n.].

FERNANDES, A. A.; TEIXEIRA, D. DE S. **Fábrica de Software: implantação e gestão de operações**. 1. ed. São Paulo, SP: Editora Atlas, 2011.

GOMES, R. **Controle de projetos: como lidar com mudanças no escopo do projeto?** Disponível em: <<https://www.mega.com.br/blog/control-de-projetos-como-lidar-com-mudancas-no-escopo-do-projeto-8393/>>. Acesso em: 14 nov. 2021.

IBM. *OpenUp*. Disponível em:

<https://download.eclipse.org/technology/epf/OpenUp/published/OpenUp_published_1.5.1.5_20121212/OpenUp/index.htm>. Acesso em: 2 maio. 2021.

LEMONIS, M. A. **Confira a entrevista exclusiva com Marcus Lemonis**. Disponível em: <<https://history.uol.com.br/noticias/confira-entrevista-exclusiva-com-marcus-lemonis>>. Acesso em: 11 dez. 2020.

MELLO, S. C. B.; LEÃO, A. L. M. DE S. Uma Revisão Crítica sobre o Uso de Diferentes Conceitos de Valor no Marketing. **RAC-Eletrônica**, abr. 2008.

MELO, A.; ABELHEIRA, R. Design Thinking & Thinking Design: Metodologia, ferramentas e uma reflexões sobre o tema. In: **Design Thinking & Thinking Design: Metodologia, ferramentas e uma reflexões sobre o tema**. 1. ed. São Paulo, SP: Novatec, 2015.

PERALTA, C. B. DA L. **Mensuração de valor em produtos inovadores na perspectiva do cliente**. Porto Alegre: UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL ESCOLA DE ENGENHARIA PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO, 2020.

PHAM, A.; PHAM, P.-V. **Scrum em ação: Gerenciamento e desenvolvimento ágil de projetos de software**. 1. ed. São Paulo, SP: Novatec, 2012.

POCIVI, V. C. B. **Um estudo para melhora do processo de ensino e aprendizagem de engenharia de software em cursos de graduação**. Recife: C.E.S.A.R – CENTRO DE ESTUDOS E SISTEMAS AVANÇADOS DO RECIFE, 2011.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. 7. ed. Porto Alegre: AMGH Editora, 2011.

RIES, E. **A startup enxuta**. 1. ed. São Paulo, SP: Leya Editora, 2012.

RODRIGUES, E. **21 Erros Clássicos da Gestão de Projetos - Eli Rodrigues - Google Livros**. 1. ed. Rio de Janeiro, RJ: Brasport, 2014.

SALVIANO, C. F. **Uma Proposta Orientada a Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de Software**. [s.l: s.n.].

SCHWABER, K.; SUTHERLAND, J. **Guia do SCRUM**. Disponível em: <https://www.Scrum.org/Portals/0/Documents/Scrum Guides/Scrum_Guide.pdf>. Acesso em: 1 maio. 2021.

SILVA, D. **O Guia do NPS: o que a Net Promoter Score tem a dizer sobre sua empresa?** Disponível em: <<https://resultadosdigitais.com.br/blog/o-que-e-nps/>>. Acesso em: 7 nov. 2021.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo, SP: Pearson Education, 2011.

SOUSA, F. C. et al. Preparação dos Acadêmicos dos Cursos de Computação da UniEVANGÉLICA para o Mercado de Trabalho através de uma Fábrica de Software

Acadêmica. **Revista Ada Lovelace**, dez. 2018.

SOUZA, M. **O que é e para que serve o Design Thinking**. Disponível em: <<https://matheusdesouza.com/2016/06/22/o-que-e-e-para-que-serve-o-design-thinking/#:~:text=Design Thinking é uma ferramenta,que os indivíduos resolvam problemas.>>. Acesso em: 2 maio. 2021.

TEBET, I. **Fábrica de software: o que é, como funciona e os benefícios para sua empresa**. Disponível em: <<https://www.objective.com.br/insights/o-que-uma-fabrica-de-software-pode-fazer-por-sua-empresa/>>. Acesso em: 11 dez. 2020.

TOLMASQUIM, E. **Aplicando o método F4P**. Disponível em: <<https://piou.com.br/2019/09/15/aplicando-o-f4p/>>. Acesso em: 7 nov. 2021.

TORRES, J. **Gestão de produtos: como aumentar as chances de sucesso do seu software**. 3ª ed. São: Casa do Código, 2015.

TORRES, J. **O que é jornada do cliente, e como criar a sua?** Disponível em: <<http://www.gyaco.com/2016/03/visao-e-estrategia-do-produto/>>. Acesso em: 23 out. 2021.

TORRES, J. **Liderança de produtos digitais: a ciência e a arte da gestão de time de produto**. São Paulo, SP: Casa do Código, 2020.

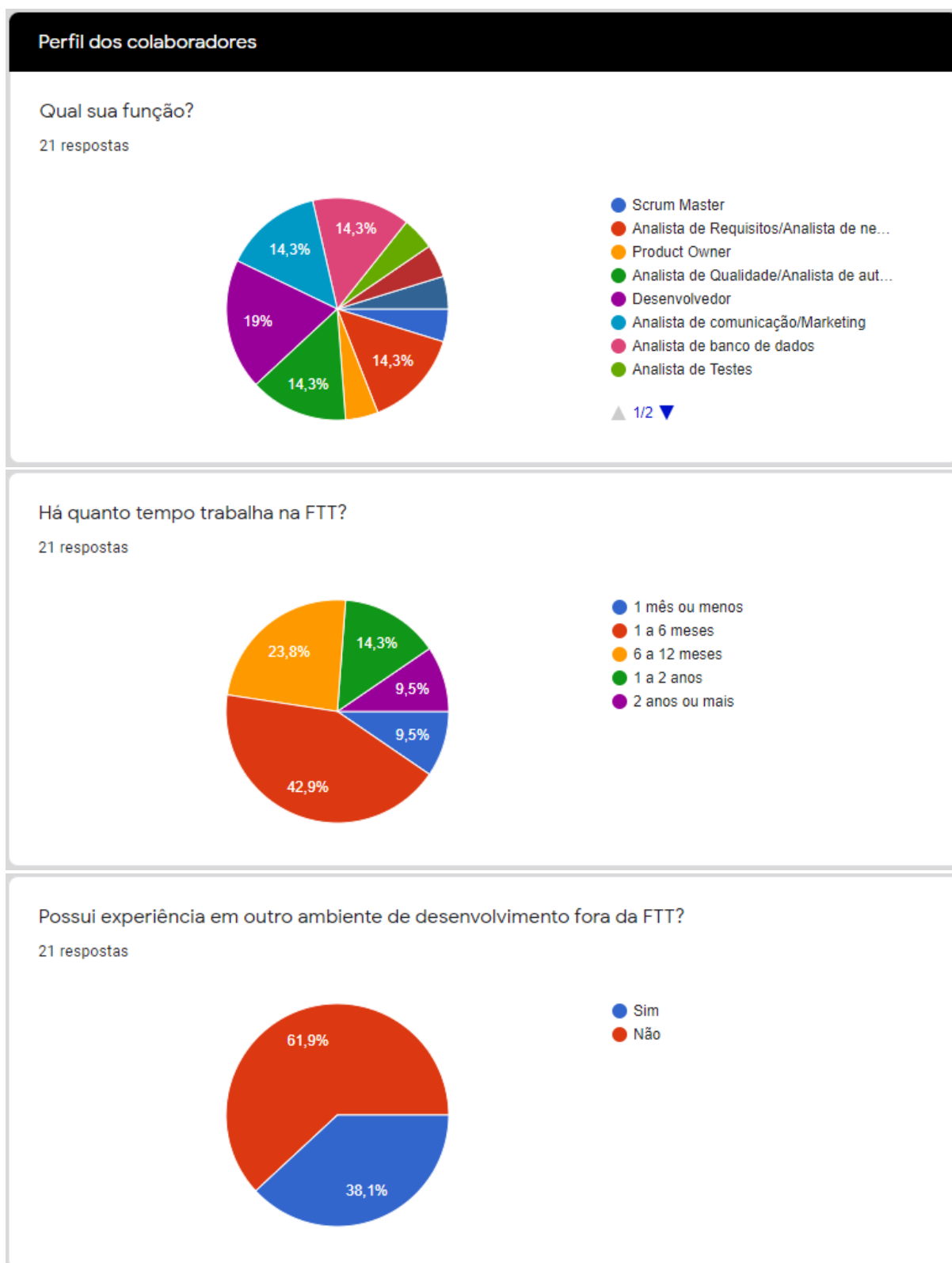
UPIS. **Design Thinking: o que é, etapas e como aplicar**. Disponível em: <<https://upis.br/blog/design-thinking/>>. Acesso em: 2 maio. 2021.

VIANNA, M. et al. **Design Thinking: Inovação em negócios**. 1. ed. Rio de Janeiro, RJ: MJV Press, 2012.

WOMACK, J.; JONES, D. **A mentalidade enxuta nas empresas**. 6. ed. Rio de Janeiro, RJ: Editora Campos, 2004.

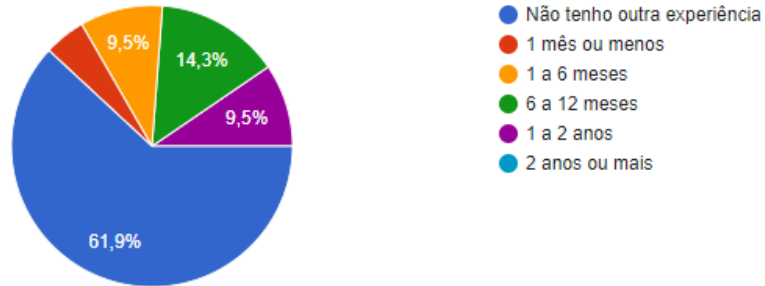
8. APÊNDICES

APÊNDICE A – Pesquisa inicial com colaboradores da FTT



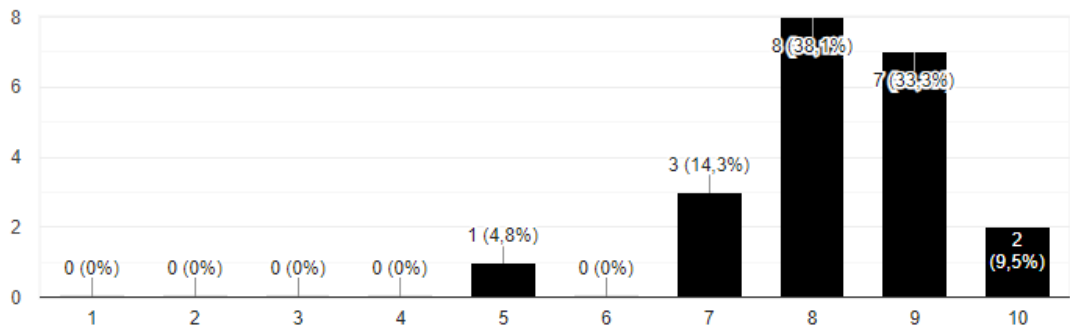
Há quanto tempo trabalhou nestas outras experiências?

21 respostas



Na sua opinião, quanto você conhece do processo de desenvolvimento utilizado pela FTT?

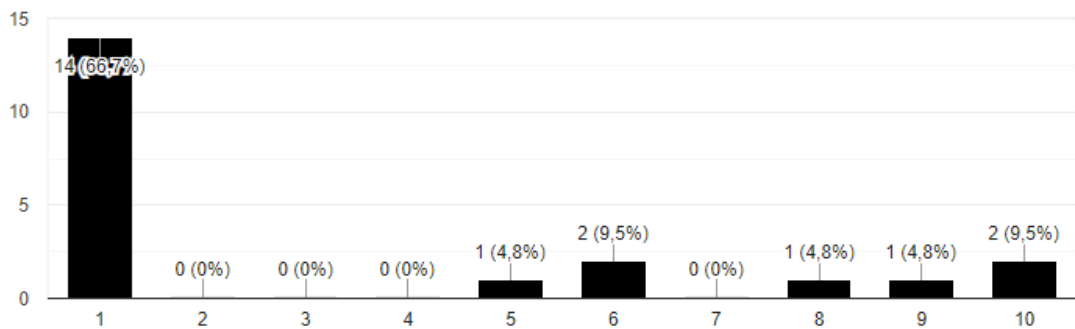
21 respostas



Processo de desenvolvimento - Envolvimento do cliente no processo

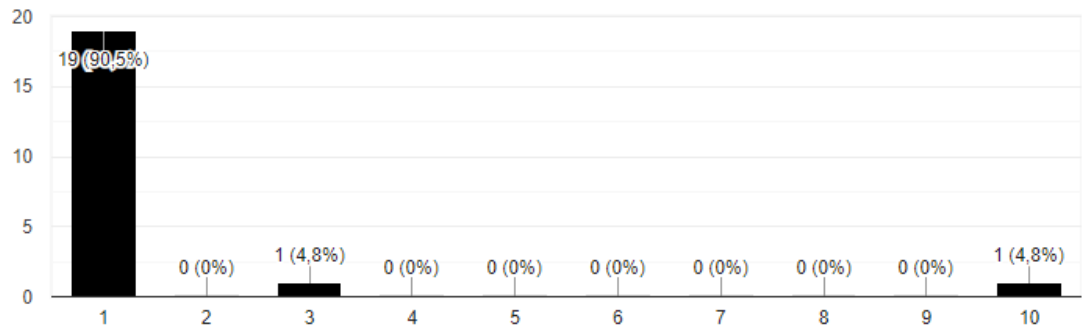
O cliente participa das reuniões de revisão do produto?

21 respostas



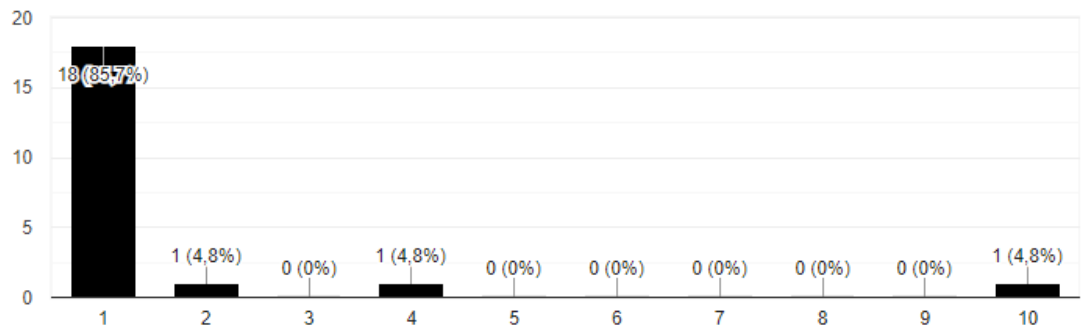
O cliente participa das reuniões diárias?

21 respostas



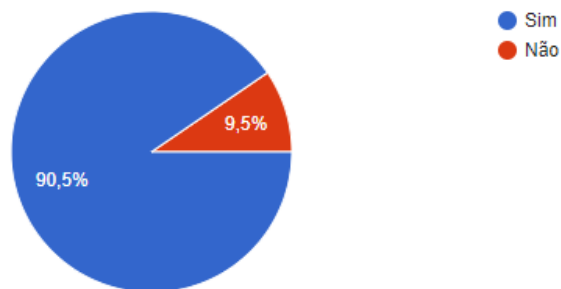
O cliente participa das reuniões planejamento (sprint ou do projeto)?

21 respostas



Você considera importante a participação do cliente no processo de desenvolvimento do software?

21 respostas





APÊNDICE B – Apresentação sobre o F4P

Fit for purpose

Framework para identificação de valor do produto

Valor do produto

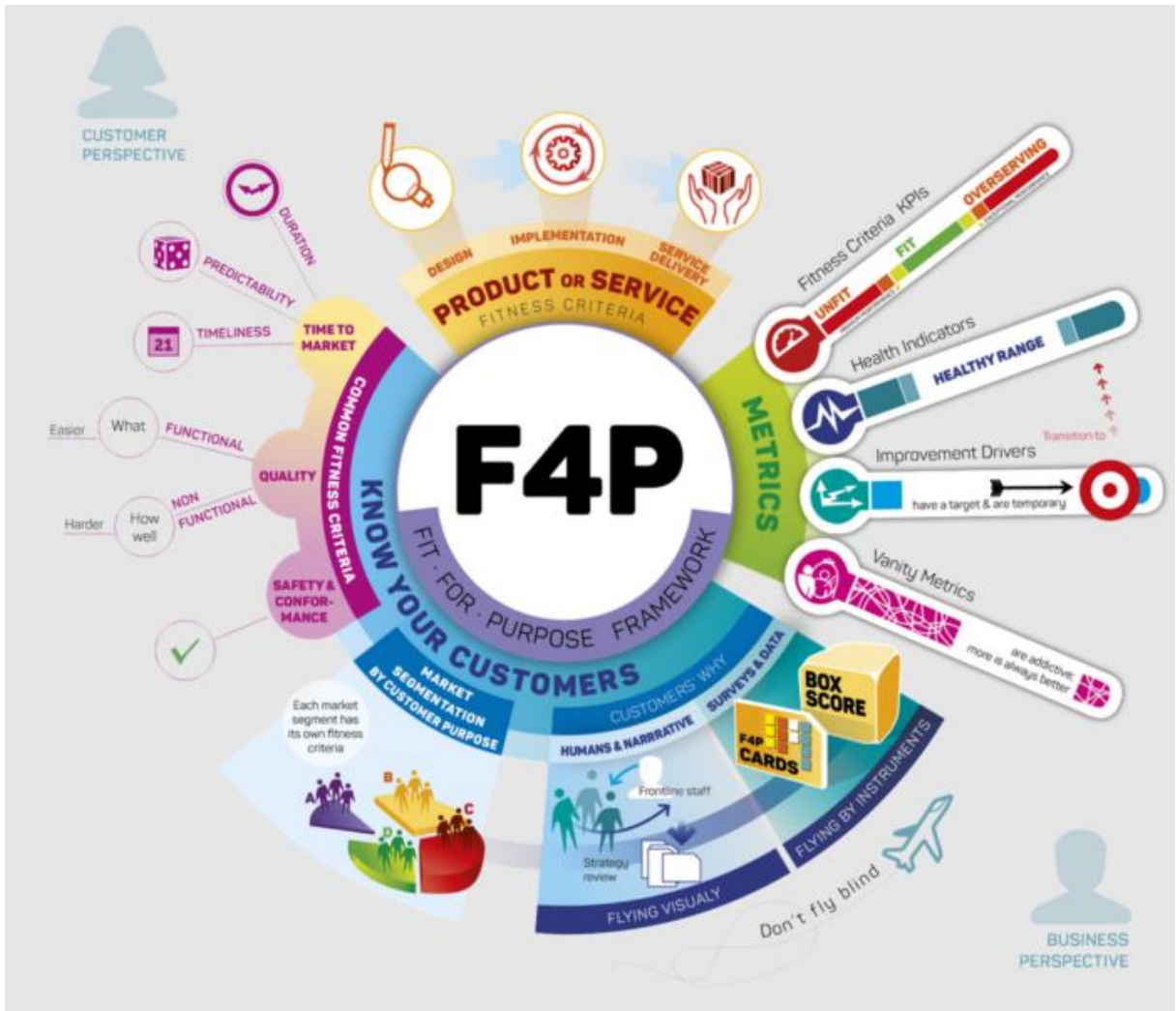
Introdução

- Produto bem construído;
- Produto que atende o propósito do cliente;
- Processo de desenvolvimento;
- Experiência do cliente.

Framework F4P

Estrutura do framework

- Processo;
- Métricas;
- Identificação de valor do produto;



- ## O que vai mudar?
- Processo;
 - Métricas;
 - KPI;
 - WIP;
 - Lead Time;
 - Quantidade de commits por sprint.
 - Identificação do valor do produto;
 - Fit box score.

Processo base definido pela Engenharia de Software			
Especificação de Requisitos	Projeto e Implementação de software	Validação de software	Evolução de software
Planejamento geral e projeto de arquitetura			
Ciclo iterativo e incremental (avaliar, selecionar, desenvolver e revisar)			Entrega do projeto
Inception			
	Elaboration	Construction	Transition
Empatia	Definição	Idealização	Prototipação
			Testar
	Design	Implementation	Service Delivery
Legenda: Scrum (Amarelo) OpenUP (Verde) Design Thinking (Vermelho) Fir for Purpose (Púrpura)			

Fit box score

- O que é?
- Como funciona?

- 1) Por que você escolheu nosso produto ou serviço?
 - () 5. Extremamente. Recebi tudo que esperava e mais um pouco.
 - () 4. Plenamente. Tudo que eu precisava foi atendido.
 - () 3. Na maior parte. Em geral foi atendido, mas algo ficou faltando.
 - () 2. Parcialmente. Alguns pontos foram ok, mas uma parte relevante deixou a desejar.
 - () 1. Um pouco. Houve itens positivos, mas a real necessidade não foi atendida.
 - () 0. Nada. Nenhuma das minhas necessidades foi atendida.
- 2) Quanto o produto ou serviço atendeu ao seu propósito?
 - () 5. Extremamente. Recebi tudo que esperava e mais um pouco.
 - () 4. Plenamente. Tudo que eu precisava foi atendido.
 - () 3. Na maior parte. Em geral foi atendido, mas algo ficou faltando.
 - () 2. Parcialmente. Alguns pontos foram ok, mas uma parte relevante deixou a desejar.
 - () 1. Um pouco. Houve itens positivos, mas a real necessidade não foi atendida.
 - () 0. Nada. Nenhuma das minhas necessidades foi atendida.
- 3) Detalhe sua resposta ao item 2.



"Don't fly blind"

David Anderson e Alexei Zheglov

APÊNDICE C – Cartões preenchidos de FBS e OKR

Fit Box Score	Escolha até três razões para você ter escolhido nossos serviço/produto.	1 - Fácil documentação rápida a ocorrências registradas pelos seguranças e usuários do produto, ressaltando a velocidade de execução; 2 – Sistema simplório e de acesso suscetível a ocasiões generalizadas. 3 – Sistema entendível ao ponto de vista do cliente, congruente a interface e representação de funcionalidade de software clara.
	Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1 2 3
	Foram superadas	X X X
	Foram totalmente atendidas	
	De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores	
	Algumas necessidades importantes não foram atendidas	
	A maior parte das minhas necessidades não foram atendidas	
Nada foi útil		
Qual foi o motivo para você escolher cada nota?	1 – Agilidade no processo da ocorrência. 2 – Coerente ao conceito do <u>app</u> simples apresentado dinamicamente. 3 – Didática entendível e sem maiores duvidas ao projeto.	

Fit Box Score	Escolha até três razões para você ter escolhido nossos serviço/produto.	1 - <u>eficiência no registro de uma ocorrência</u> 2 - _____ 3 - _____
	Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1 2 3
	Foram superadas	
	Foram totalmente atendidas	X
	De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores	
	Algumas necessidades importantes não foram atendidas	
	A maior parte das minhas necessidades não foram atendidas	
Nada foi útil		
Qual foi o motivo para você escolher cada nota?	1 - <u>devido a razão ter sido atendida de uma forma como desejado</u> 2 - _____ 3 - _____	

Fit Box Score

Escolha até três razões para você ter escolhido nossos serviço/produto.

1 - Necessidade de agilizar o processo de registro de ocorrências.
 2 - _____
 3 - _____

Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1	2	3
<i>Foram superadas</i>			
<i>Foram totalmente atendidas</i>	X		
<i>De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores</i>			
<i>Algumas necessidades importantes não foram atendidas</i>			
<i>A maior parte das minhas necessidades não foram atendidas</i>			
<i>Nada foi útil</i>			

Qual foi o motivo para você escolher cada nota?

1 - O serviço é confiável e performa como esperado.
 2 - _____
 3 - _____

Fit Box Score

Escolha até três razões para você ter escolhido nossos serviço/produto.

1 - A praticidade ao registrar uma ocorrência
 2 - A agilidade para obter informações sobre ocorrências
 3 - Para melhor gerenciamento e compreensão das ocorrências

Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1	2	3
<i>Foram superadas</i>		X	X
<i>Foram totalmente atendidas</i>	X		
<i>De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores</i>			
<i>Algumas necessidades importantes não foram atendidas</i>			
<i>A maior parte das minhas necessidades não foram atendidas</i>			
<i>Nada foi útil</i>			

Qual foi o motivo para você escolher cada nota?

1 - A nota foi atribuída pelo fato de ser bastante esclarecedor e simples
 2 - A nota foi atribuída pelo fato de obter o ícone de notificações que nos avisa quando surgir novas ocorrências
 3 - A nota foi atribuída pelo fato de ser bastante simples, prático e fácil de se aprender

Fit Box Score

Escolha até três razões para você ter escolhido nossos serviço/produto.	1 - <u>Facilitar o processo de gerenciamento de ocorrências.</u> 2 - <u>Agilizar o registro de ocorrências.</u> 3 - _____
Quanto você classifica que cada motivo desse foi atendido com o nosso produto	1 2 3
<i>Foram superadas</i>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
<i>Foram totalmente atendidas</i>	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
<i>De modo geral foram atendidas, mas deixaram a desejar em alguns pontos menores</i>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
<i>Algumas necessidades importantes não foram atendidas</i>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
<i>A maior parte das minhas necessidades não foram atendidas</i>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
<i>Nada foi útil</i>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Qual foi o motivo para você escolher cada nota?	1 - <u>Porque o sistema é eficaz no gerenciamento das ocorrências.</u> 2 - <u>O sistema desempenhas bem a função de registrar a ocorrência.</u> 3 - _____

Objective and Key Results - OKR

Obj - Ser reconhecida como uma empresa referência na consolidação de profissionais para o mercado de trabalho.

KR¹ - Ter 80% dos alunos egressos inseridos no mercado de trabalho em 2021/2.

KR² - Estruturar ao menos duas parcerias com empresas locais em 2021/2.

KR³ - Ter uma retenção dos alunos egressos de 2021/2 de pelo menos 3 meses.

Key Performance Indicator- KPI

1 - Reduzir em 60% o número de alunos desistentes.

2 - Aumentar em 100% o número de alunos participantes da capacitação em relação ao semestre anterior.

3 - Elevar a nota de satisfação da empresa para 8 pontos numa escala de 0 a 10.