

FATEC
Santo André
Eletrônica Automotiva

Alessandro Baraldi

**PONTO DE VISTA: Aplicação do *framework* SCRUM na indústria
automotiva**

Santo André - SP
2020

Alessandro Baraldi

PONTO DE VISTA: Aplicação do *framework* SCRUM na indústria automotiva

Trabalho de Conclusão de Curso apresentado ao Curso Superior em Eletrônica Automotiva da FATEC Santo André, orientado pelo Prof. Me. Cleber William Gomes, como requisito parcial para obtenção do título de tecnólogo em Eletrônica Automotiva.

Santo André - SP

2020

FICHA CATALOGRÁFICA

B224p

Baraldi, Alessandro

Ponto de vista: aplicação do framework scrum na indústria automotiva / Alessandro Baraldi. - Santo André, 2020. – 95f: il.

Trabalho de Conclusão de Curso – FATEC Santo André.
Curso de Tecnologia em Eletrônica Automotiva, 2020.

Orientador: Prof. Me. Cleber William Gomes

1. Gerenciamento de projetos. 2. Indústria automotiva. 3. Ferramenta. 4. Processos. 5. Framework Scrum. 6. Produtividade. 7. Produto. 8. Mercado. I. Ponto de vista: aplicação do framework scrum na indústria automotiva.

658.404

Alessandro Baraldi

PONTO DE VISTA: Aplicação do *framework* SCRUM na indústria automotiva

Trabalho de Conclusão de Curso apresentado a FATEC SANTO ANDRÉ como requisito parcial à obtenção de título de Tecnólogo em Eletrônica Automotiva.

BANCA EXAMINADORA

Local: Fatec Santo André

Horário: 18:00

Data: 15/07/2020

Prof. Me. Cleber William Gomes
Presidente da Banca
Fatec Santo André

Prof. Me. Fernando Garup Dalbo
Primeiro membro da Banca
Fatec Santo André

Prof. Me. Orlando de Salvo Junior
Segundo Membro da Banca
Fatec Santo André

Santo André - SP

2020

Gostaria de dedicar este trabalho a todas as pessoas que procuram por conhecer e adquirir mais conhecimento sobre o uso da ferramenta de gerenciamento de projetos *framework* Scrum para a condução de projetos, podendo ser aplicado em qualquer área de atuação em que existam pessoas, as quais precisam trabalhar em conjunto para conseguir alcançar um objetivo comum.

AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus por ter me dado saúde e força para superar todas as dificuldades que surgiram durante todos esses anos. A Faculdade de Tecnologia de Santo André (FATEC), seu corpo docente, direção e administração que sempre me auxiliaram em todos os momentos bons e difíceis desse ciclo que termina com a conclusão deste trabalho. Agradeço a todos os professores por me proporcionarem o conhecimento não apenas racional, mas a manifestação do caráter e afetividade da educação no processo de formação profissional, por tanto que se dedicaram a mim, não somente por terem me ensinado, mas por terem me feito aprender a ser um profissional. Os professores dedicados terão os meus eternos agradecimentos. Ao professor Me. Cleber William Gomes, pela orientação, apoio, confiança, pelo empenho dedicado à elaboração deste trabalho. Gostaria de agradecer ao professor Me. Fernando Garup Dalbo pelo auxílio, motivação, experiência, apoio e confiança durante todas as orientações e conversas até o final do desenvolvimento deste projeto de conclusão para obtenção do título de tecnólogo. Agradeço a minha mãe Maria de Lourdes Cunha Paredes, heroína que me deu apoio, incentivou nas horas difíceis, de desânimo, cansaço e pelo amor. Obrigado minhas irmãs Cristiane Baraldi e Natali Baraldi, que nos momentos de minha ausência dedicados ao estudo superior, sempre fizeram entender que o futuro é feito a partir da constante dedicação no presente. Meus agradecimentos aos amigos, companheiros de trabalhos, amizade que faz parte da minha formação e que vai continuar presentes em minha vida com certeza.

EPÍGRAFE

“Democracia é oportunizar a todos o mesmo ponto de partida. Quanto ao ponto de chegada, depende de cada um.”
FERNANDO SABINO

“Ó, vida futura! Nós te criaremos.”
CARLOS DRUMMOND DE ANDRADE

“Liberdade é uma palavra que o sonho humano alimenta, não há ninguém que explique e ninguém que não entenda.”
CECÍLIA MEIRELES

“Maravilhas nunca faltaram ao mundo; o que sempre falta é a capacidade de senti-las e admirá-las.”
MARIO QUINTANA

“Não queremos perder, nem deveríamos perder: saúde, pessoas, posição, dignidade ou confiança. Mas perder e ganhar faz parte do nosso processo de humanização.”
LYA LUFT

RESUMO

Este trabalho projeta apresentar elementos técnicos relacionados ao gerenciamento de projetos objetivando ser uma fonte de informações qualitativa exploratória e descritiva sobre o *framework* Scrum na Indústria automotiva. A pesquisa se caracteriza como qualitativa exploratória e descritiva, pois este estudo visa à compreensão, entendimento e precisão do desenvolvimento de todos os eventos do Scrum de modo que traga informações para melhor utilização desta ferramenta de gestão de gerenciamento de projetos. Apresentando que quando bem aplicado os papéis, eventos, artefatos e regras do *framework* Scrum, estas práticas funcionam de forma natural trazendo diversos benefícios para toda a organização e disseminando o conhecimento para todos os colaboradores da empresa. Com a 4ª Revolução Industrial, as empresas estão gerando cada vez mais dados que precisam ser processados, visualizados, analisados e interpretados de forma rápida e eficiente para aprimorar o processo de tomada de decisões, gerando um grande impacto no conceito de produtividade e na competitividade no mercado consumidor. Nesse cenário, o uso do Scrum permite a melhora do acompanhamento e desenvolvimento de qualquer projeto. Como foco central, serão apresentados os atos existentes no *framework* Scrum que proporcionam a eficácia de suas práticas de gerenciamento de produto e técnicas de trabalho, de modo que seja possível estar sempre melhorando o produto, o time e o ambiente de trabalho, e utilizando uma abordagem iterativa e incremental para melhorar a previsibilidade e o controle de riscos.

Palavras-chaves: *Framework*. Scrum. Projeto. Gerenciamento. Indústria. Ferramenta ágil. Automotiva.

ABSTRACT

This project intends to present technical elements related to project management aiming to be a source of qualitative exploratory and descriptive information about the Scrum framework in the automotive industry. The research is characterized as exploratory and descriptive qualitative, because this study aims the understanding and accuracy of the development of all Scrum events in order to bring information for better use of this project management tool. Presenting that the roles, events, artifacts and rules of the Scrum framework are well applied, these practices work in a natural way, bringing several benefits to the whole organization and disseminating knowledge to all the company's employees. With the 4th Industrial Revolution, companies are generating more and more data that need to be processed, visualized, analyzed and interpreted in a fast and efficient way to improve the decision making process, generating a great impact on the concept of productivity and competitiveness in the consumer market. In this scenario, the use of Scrum allows the improvement of the follow-up and development of any project. As a central focus, the existing acts in the Scrum framework that provide the effectiveness of its product management practices and work techniques will be presented, so that it is always possible to improve the product, the team and the work environment, and using an iterative and incremental approach to improve predictability and risk control.

Keywords: Framework. Scrum. Project. Management. Industry. Agile tool. Automotive.

LISTA DE ABREVIATURAS E SIGLAS

IoT	<i>Internet of Things</i>
PMI	<i>Project Management Institute</i>
FDD	<i>Feature-Driven Development</i>
MSF	<i>Microsoft Solutions Framework</i>
DSDM	<i>Dynamic Systems Development Methodology</i>
XP	<i>Extreme Programming</i>
M2M	<i>Machine to Machine</i>
CPS	<i>Cyber-Physical Systems</i>
RFID	<i>Radio Frequency Identification</i>
PwC	<i>Price water house Cooper</i>
MVP	<i>Minimum Viable Product</i>
MMF	<i>Minimally Marketable Feature</i>
IEDI	Instituto de Estudos para o Desenvolvimento Industrial
ABDI	Agência Brasileiros de Desenvolvimento Industrial
CNI	Confederação Nacional da Indústria
UNCTAD	Conferência das Nações Unidas sobre Comércio e Desenvolvimento
EPI	Equipamento de Proteção Individual
TPS	<i>Toyota Production System</i>
PBI	<i>Product Backlog Items</i>
SIL	<i>Software in the Loop</i>
MIL	<i>Model in the Loop</i>

LISTA DE ILUSTRAÇÕES

Figura 1 - Visão geral simplificada dos eventos e ciclos do Scrum.....	22
Figura 2 - Representação dos integrantes do Scrum.	23
Figura 3 - O Scrum dispõe de uma equipe de desenvolvimento em seu núcleo.....	24
Figura 4 - Sequência de Sprints.	25
Figura 5 - Representação em um quadro real do Sprint Backlog.	26
Figura 6 - Ferramenta gráfica que demonstra a evolução do Sprint.	27
Figura 7 - Representação dos eventos da Sprint do framework Scrum.	28
Figura 8 - A base fundamental do framework Scrum.....	30
Figura 9 - Exemplo de evolução de um MVP.	32
Figura 10 – Ilustração de Stakeholders.	34
Figura 11 - Orientação das partes interessadas (stakeholders).	35
Figura 12 - O Framework Nexus™, exoesqueleto do Scrum Escalado.	36
Figura 13 - Membros da equipe de integração do Nexus.	37
Figura 14 - Estrutura do Time de integração do Nexus.	37
Figura 15 – Sequência para realização da Nexus Sprint Planning.	41
Figura 16 - Modelo do <i>framework Waterfall</i>	46
Figura 17 - Representação das 4 revoluções industriais.	49
Figura 18 - Evolução da indústria.	50
Figura 19 - Representação dos nove pilares da indústria 4.0.	52
Figura 20 - Modelo de Ciclo-V de desenvolvimento.	65
Figura 21 - A abordagem tradicional de desenvolvimento do setor automotivo.	66
Figura 22 – Os fatores para a implementação ágil na indústria automotiva.	70
Figura 23 - Projeto Collectivo.	75
Figura 24 - MAN Truck & Bus apresenta o conceito de caminhão elétrico CitE.....	76
Figura 25 - Exemplo de organização do Squad.....	79
Figura 26 – Composição dos Squads em desenvolver funcionalidades específicas.	80
Figura 27 - Exemplo de vários Squads de uma Tribo.....	81
Figura 28 - Exemplo de Tribo que forma um Chapter.....	82
Figura 29 - Exemplo de Guilds entre Tribes.	83

LISTA DE TABELAS

Tabela 1 - Apresenta as práticas do Scrum.	29
Tabela 2 - Termos utilizados no Framework Scrum.	31
Tabela 3 - Fluxo de processo do Nexus framewrok Scrum.	39

SUMÁRIO

1. INTRODUÇÃO.....	14
1.1 OBJETIVO.....	14
1.2 JUSTIFICATIVA.....	14
1.3 MOTIVAÇÃO.....	15
1.4 ESTRUTURA DO TRABALHO.....	15
2. FUNDAMENTAÇÃO TEÓRICA.....	17
2.1 MÉTODOS ÁGEIS (<i>SOFTWARE</i>).....	19
2.2 <i>FRAMEWORK</i> SCRUM.....	21
2.2.1 <i>Product Owner</i>	22
2.2.2 <i>Scrum Master</i>	23
2.2.3 <i>Scrum Team</i> ou Time de Desenvolvimento.....	23
2.2.4 <i>Product Backlog</i>	24
2.2.5 <i>Sprint Planning</i>	25
2.2.6 <i>Sprint Backlog</i>	25
2.2.7 <i>Sprint</i>	26
2.2.8 <i>Daily Scrum</i>	27
2.2.9 <i>Sprint Review</i>	28
2.2.10 <i>Sprint Retrospective</i>	28
2.3 DEFINIÇÕES.....	29
2.3.1 MVP (<i>Minimum Viable Product</i>).....	32
2.3.2 <i>Stakeholders</i>	33
2.4 <i>NEXUS FRAMEWORK</i> SCRUM PARA ESCALONAMENTO ÁGIL.....	35
2.4.1 Fluxo de processo do <i>framework Nexus</i>	38
2.5 OS EVENTOS DO <i>FRAMEWORK NEXUS</i>	40
2.5.1 A Reunião de Planejamento do <i>Nexus</i>	40
2.5.2 Reunião Diária do Scrum no <i>Nexus</i>	41
2.5.3 Revisão da <i>Sprint</i> do <i>Nexus</i>	41
2.5.4 Retrospectiva da <i>Sprint</i> do <i>Nexus</i>	41
2.5.5 Refinamento.....	42
2.6 OS ARTEFATOS DO <i>FRAMEWORK NEXUS</i>	43

2.6.1	<i>Backlog</i> do Produto.....	43
2.6.2	Meta do <i>Nexus</i>	43
2.6.3	<i>Backlog</i> da <i>Sprint</i> do <i>Nexus</i>	44
2.6.4	Incremento Integrado	44
2.7	MODELO CASCATA (<i>FRAMEWORK WATERFALL</i>)	44
2.7.1	As vantagens do uso do Modelo Cascata (<i>framework Waterfall</i>).....	46
2.7.2	As desvantagens do uso do Modelo Cascata (<i>framework Waterfall</i>).....	47
2.8	INDÚSTRIA 4.0	48
2.8.1	Evolução industrial 1.0 a 4.0	49
2.8.2	Pilares da indústria 4.0.....	51
2.8.3	Indústria 4.0 no mundo	54
2.8.4	Indústria 4.0 no Brasil	56
3	METODOLOGIA.....	59
3.1	QUAL RELAÇÃO ENTRE <i>FRAMEWORK SCRUM</i> E INDÚSTRIA 4.0.....	59
3.1.1	As vantagens do uso do <i>framework Scrum</i>	61
3.1.2	As desvantagens do uso do <i>framework Scrum</i>	64
3.2	COMO É O GERENCIAMENTO CONVENCIONAL NA INDÚSTRIA AUTOMOTIVA (HOJE).....	64
3.2.1	As vantagens do uso do Modelo em V	67
3.2.2	As desvantagens do uso do Modelo em V	67
3.3	COMO INTRODUZIR O SCRUM NO DESENVOLVIMENTO ÁGIL NA INDÚSTRIA AUTOMOTIVA	68
3.3.1	Exemplo de aplicação do <i>framework Scrum</i> no segmento automotivo.....	71
3.4	EMPRESAS QUE JÁ USAM / USAVAM O SCRUM.....	76
3.4.1	O modelo Spotify <i>Squads</i>	78
4	CONCLUSÃO.....	84
5	REFERÊNCIAS BIBLIOGRÁFICAS.....	86

1. INTRODUÇÃO

O avanço tecnológico é um processo que está cada vez mais comum na vida das indústrias, das pessoas e influenciará todas as áreas da economia, provocando diversas transformações econômicas e sociais nos próximos anos. O presente trabalho tem o objetivo de apresentar informações sobre o tema “Aplicação do *framework* Scrum na indústria automotiva”. Devido ao avanço extraordinário da digitalização e que continua em constante evolução no mundo dos negócios, diversas empresas começaram a concentrar cada vez mais no desenvolvimento de projetos como solução para adquirir uma vantagem para manter-se competitiva no mercado dos negócios, uma vez que os projetos são os pontos iniciais das transformações.

Neste contexto, visando melhorias no desenvolvimento de projetos, o uso do *framework* Scrum como ferramenta ágil para condução do gerenciamento de projetos tem como objetivo de conduzir todo o ciclo de desenvolvimento do projeto e podendo ser aplicado em qualquer contexto em que existam pessoas, as quais necessitam trabalhar juntas para obter um objetivo comum.

1.1 Objetivo

O objetivo principal deste trabalho é ser uma fonte de informações qualitativa exploratória e descritiva sobre o *framework* Scrum na Indústria automotiva, sendo uma ferramenta ágil para gerir e planejar projetos de *software* e *hardware*, e introduzi-lo no conceito de indústria 4.0 que envolve as inovações tecnológicas nos campos da automação industrial, definindo sistemas de produção inteligentes, conectando máquinas, sistemas e ativos.

1.2 Justificativa

Nos dias de hoje, cada vez mais as indústrias buscam por eficiência e produtividade nos projetos de desenvolvimento de *software*. Sabe-se que a utilização de boas práticas para embasar o gerenciamento de um projeto faz com que otimize tempo de desenvolvimento. Um dos desafios na área está na dificuldade de se compreender os reais problemas e criar soluções que efetivamente atendam aos propósitos dos clientes.

O tema proposto do projeto foi escolhido com a justificativa principal de apresentar o *framework* Scrum como uma ferramenta de gerenciamento de projetos e desenvolvimento de *softwares*. Surge a necessidade de entender como funciona o desenvolvimento dos projetos para que possamos determinar a viabilidade do *framework* em uma determinada aplicação. Todas as empresas utilizam um modelo ou método de gestão de projetos de *software*, e possui uma cultura organizacional única e que diferencia das outras e buscam cumprir objetivos por meio da produção de entregas.

1.3 Motivação

Existem razões acadêmicas que motivam buscar informações para melhor compreensão das ferramentas de gestão de gerenciamento de projetos (Características, objetivos, filosofia, boas praticas, métodos) com o objetivo de adquirir conhecimento técnico de como aplicar, desenvolver e proporcionar aprendizado sobre o assunto. Existem razões práticas que motivam buscar escolher uma metodologia, técnica, *framework* ou parte de uma determinada aplicação de uma organização que seja viável para ser estudada e aplicada na realidade em que vivemos.

Não podemos considerar separadamente as duas razões pelo fato que uma vez que os estudos acadêmicos auxiliam nas escolhas práticas, as razões práticas influenciam nos critérios em que aplicamos os estudos acadêmicos. Assim, não podemos analisá-las separadamente, ambas estão relacionadas e são responsáveis por formarem as qualificações profissionais do indivíduo que deseja buscar por mais preparo profissional.

1.4 Estrutura do trabalho

O trabalho está dividido em cinco capítulos. Na fundamentação teórica será apresentada a pesquisa literária, conceitos analisados e o desenvolvimento do *framework* Scrum, *Nexus framework* Scrum e o *framework* Waterfall. O gerenciamento de projeto e ferramentas ágeis é fundamental para estruturarmos a ideia de como desenvolver projetos, exige não só conhecimentos técnicos voltados a área de desenvolvimento, mas requer a necessidade de aprimorar seus

conhecimentos na gestão de projetos, utilizando técnicas e processos para gerenciar melhor seus trabalhos e aumentar as suas chances de sucesso. Sobre a Indústria 4.0 apresentaremos a evolução industrial, onde houve quatro revoluções industriais durante a história da humanidade, e na atualidade nos encontramos na quarta revolução industrial, sendo nomeada de Indústria 4.0. Apresentaremos os pilares fundamentais da Indústria 4.0, como está sendo levado em conta a Indústria 4.0 no Brasil e no mundo.

No capítulo metodologia serão apresentados e exemplificados sobre os métodos usados para o desenvolvimento do projeto. Será descrito qual é a relação entre a Indústria 4.0 e o Scrum, exemplos de introduzir o *framework* Scrum no gerenciamento de projetos automotivos com a presença da ideia de Indústria 4.0, apresentar quais seriam as vantagens e desvantagens da utilização do Scrum. Também será apresentado como é o gerenciamento convencional na indústria automotiva nos dias de hoje, e suas vantagens e desvantagens para desenvolvimento de *software*. Exemplificar com empresas que já usaram ou usam o Scrum em outros segmentos, apresentando um case de sucesso como o Spotify.

2. FUNDAMENTAÇÃO TEÓRICA

O PMI (2017) define projeto como um esforço temporário empreendido para criar um produto, serviço ou resultado único. Ou seja, um projeto tem como finalidade produzir algo que visa satisfazer uma necessidade, podendo ser um produto físico, serviço ou resultados que tenham particularidades únicas em relação a todas as outras opções.

Os projetos são uma maneira chave de criar valor e benefícios nas organizações. No ambiente de negócios atual, os líderes organizacionais precisam ser capazes de gerenciar orçamentos cada vez mais apertados, prazos mais curtos, recursos mais escassos e uma tecnologia que muda rapidamente. O ambiente de negócios é dinâmico, com um ritmo acelerado de mudança. Para se manterem competitivas na economia mundial, as empresas estão adotando o gerenciamento de projetos para entregar valor de negócio de forma consistente. (PMI, 2017, p. 46).

Implementar um projeto exige um processo de gerenciamento. A definição de Gerenciamento de Projetos é conceituada de maneiras diferentes na literatura. Mas, em geral, as definições convergem. O PMI (2017) define que gerenciamento de projetos é a capacidade de aplicar conhecimento, habilidade, ferramentas e técnicas na realização das atividades do projeto a fim de cumprir os objetivos estabelecidos e atender seus requisitos, com isso, sendo possível aplicar e integrar de forma apropriada os processos de gerenciamento de projetos identificados para o projeto desenvolvido, assim, permite que as organizações possam executar os projetos de forma eficaz e eficiente.

A proposta de um *framework* são ideias não expressas de maneira explícita de como solucionar um determinado problema. Com uma abordagem genérica permite que o desenvolvedor possa dedicar seus esforços na resolução do problema. Wong e Aspinwall (2004) relatam que *framework* é como um conjunto de pressupostos básicos ou princípios fundamentais que formam as bases para uma determinada ação. Auxiliando diretamente nos processos de tomada de decisões e de resoluções de problemas.

D'Avillar (2019) relata que o *Framework* é um conjunto de técnicas, ferramentas ou conceitos pré-definidos usados para resolver um problema de um projeto ou domínio específico. É, basicamente, uma estrutura de trabalho que atua com funções pré-estabelecidas que se adaptam a situação e a organização em questão.

Guedes (2019) relata que um *framework* é uma estrutura-base que contém um conjunto de funções e componentes pré-definidos, funções e componentes estes

que se relacionam para disponibilizar funcionalidades específicas ao desenvolvimento de software. Estas funções e componentes genéricos pré-prontos agilizam o processo, poupam tempo e evitam retrabalho para o desenvolvedor.

Cruz (2013) relata que Scrum não é uma sigla, mas sim o nome de uma das jogadas mais conhecidas do esporte conhecido como rúgbi. Os jogadores disputam a reposição de bola, e é necessária a participação de todos os jogadores do time atuando em conjunto no mesmo objetivo, sendo que se um deles falhar todos falham. Este trabalho em equipe é bem caracterizado no *framework* do Scrum, e por isso o seu nome foi originado desta palavra.

Sabbaght (2013) relata que *framework* ou arcabouço é uma estrutura básica que pretende servir de suporte e guia para a construção, a partir da expansão dessa própria estrutura, de algo com uso prático. Enquanto *framework*, o Scrum não define práticas específicas e detalhadas a serem seguidas. Scrum não prescreve, por exemplo, como o time deve desenvolver impedimentos, nem como devem ser levantadas as necessidades de negócios ou se deve lidar com os clientes do projeto.

Schwaber (2017) relata que Scrum é um *framework* estrutural que está sendo usado para gerenciar o trabalho em produtos complexos desde o início de 1990. Scrum não é um processo, técnica ou um método definitivo. Em vez disso, é um *framework* dentro do qual você pode empregar vários processos ou técnicas. O Scrum deixa clara a eficácia relativa de suas práticas de gerenciamento de produto e técnicas de trabalho, de modo que você possa continuamente melhorar o produto, o time e o ambiente de trabalho.

Franco (2007) relata que Scrum não define uma técnica específica para o desenvolvimento de *software* durante a etapa de implementação. Ele se concentra em descrever como os membros das equipes devem trabalhar para produzir um sistema flexível, num ambiente de mudanças constantes.

Sabbaght (2013) define que devido as constantes mudanças oriundas do mercado profissional, o gerente de projetos tem a necessidade de se adaptar frequentemente a este tipo de situação, dessa forma, precisando estar reorganizando suas decisões estratégicas, reduções de custos operacionais e adaptar seus processos e serviços de forma a atender as demandas dos clientes. Com base nesta situação, confirma-se a importância de uma forma de gerenciamento de projeto mais ágil e flexível às mudanças. Com isso, o uso do *framework* Scrum pode se apresen-

tar como um diferencial, visto que preza pelo desenvolvimento interativo e incremental e a flexibilidade exigida por causa das frequentes mudanças impostas pelo mercado de trabalho, além de ser capaz de maximizar a vantagem competitiva.

O Scrum é um *framework* de desenvolvimento ágil que busca melhorias no período de desenvolvimento de projetos, buscando a satisfação do cliente. Assim, trazendo melhorias no processo de comunicação e na interação entre os envolvidos no projeto, promovendo um constante *feedback* durante o processo de construção do projeto. No Scrum o processo de comunicação é feito de forma colaborativa e direta entre os envolvidos no projeto através das reuniões diárias, na revisão das *sprints* e em todo processo de desenvolvimento do projeto. Rodrigues (2017) relata que o Scrum se sobressai diante dos demais métodos ágeis pelo fato de dar maior destaque na gestão de projetos, agregando atividades de monitoramento, *feedback's* através de reuniões rápidas e diárias com toda a equipe, objetivando a identificação e correção de quaisquer falhas ou impedimentos, que possam surgir durante o processo de desenvolvimento.

2.1 Métodos ágeis (*Software*)

Os métodos ágeis passaram a ser uma alternativa aos métodos tradicionais de gerenciamento de projetos e têm ajudado muito no processo de desenvolvimento, auxiliando as equipes encararem a diversidade e imprevisibilidades que há ao longo de um projeto através de entregas incrementais e ciclos iterativos. Bernardo (2015) relata que os métodos ágeis são uma alternativa à gestão tradicional de projetos, eles nasceram nos braços do desenvolvimento de *software*, mas hoje podem ser aplicados a qualquer tipo de projeto inclusive os que não se remetem ao *software*.

Brasileiro (2017) relata que o modelo de entrega ágil é baseado em ciclos iterativos e incrementais, o que traz flexibilidade e adaptabilidade. Uma característica importante é a inspeção e adaptação dos ciclos e iterações, focados em gerar melhoria contínua para as equipes e processos.

Silva (2009) relata que *Feature-Driven Development* (FDD) é uma metodologia de desenvolvimento de *software* que inclui alguns benefícios de processos rigorosos, como modelagem, planejamento prévio e controle do projeto, assim como

contém características de processos ágeis, como foco na programação, interação constante com o cliente e entrega frequente de versão do produto.

O *Feature Driven Development* (FDD) é definido ao redor de um conjunto central de práticas que, embora não sejam inovadoras, são utilizadas de forma inovadora; elas reforçam e complementam umas as outras. A equipe pode implementar somente algumas das práticas, mas se não considerar todas elas não obterá todos os benefícios previstos pelo FDD. O FDD poder ser adaptado segundo a forma de trabalhar de cada equipe e a experiência das pessoas. É esta flexibilidade que torna o FDD fácil de ser implementado e utilizado. Contudo, é aconselhável que todas as práticas sejam consideradas. (MARTINS, 2007, p. 355).

Camargo (2018) relata que a metodologia ágil XP (*Extreme Programming*). É dirigido ao desenvolvimento de *softwares*, realizado a partir de três pilares: agilidade no desenvolvimento da solução, economia de recursos e qualidade do produto final. Mas para chegar à excelência dos serviços, esse método também é focado em valores como comunicação, simplicidade, *feedback*, coragem e respeito.

Vasconcelos (2006) relata que outra particularidade de *Extreme Programming* (XP) é a preferência por *software* livres e/ou código aberto. Apesar de não ser mencionado esta obrigatoriedade em qualquer referência, sempre encontramos indicações de *software* livres e código aberto como Wiki e CVS em suas publicações, e as ferramentas concebidas para projetos XP como o *XPlanner* são em grande maioria neste contexto.

Existe uma categoria de processos de desenvolvimento conhecida como Processos Ágeis de Desenvolvimento, dentro da qual XP e outros processos se encaixam. Eles compartilham a premissa de que o cliente acaba aprendendo sobre suas necessidades na medida em que é capaz de manipular o sistema que está sendo produzido. Com base no *feedback* do sistema, ele reavalia as suas necessidades e prioridades, gerando mudanças que devem ser incorporadas ao *software*. O aprendizado é importante porque permite que o cliente direcione o desenvolvimento de modo que a equipe produza sempre aquilo que tem o maior valor para o seu negócio. O XP é um processo de desenvolvimento que busca garantir que o cliente receba o máximo de valor de cada dia de trabalho da equipe de desenvolvimento. Ele é organizado em torno de um conjunto de valores e práticas que atuam de forma harmônica em torno de um conjunto de valores e práticas que atuam de forma harmônica e coesa para assegurar que o cliente sempre receba um alto retorno do investimento em *software*. (TELES, 2014, p. 24).

Roque (1998) relata que *Microsoft Solutions Framework* (MSF) é uma biblioteca de modelos, conceitos e orientações para construção e utilização de sistemas empresariais. MSF busca auxiliar os clientes da Microsoft a alinhar seus objetivos tecnológicos e negócio, reduzir os custos dos ciclos de vida usando novas tecnologias através das seguintes funções: planejamento; desenvolvimento; gerenciamento.

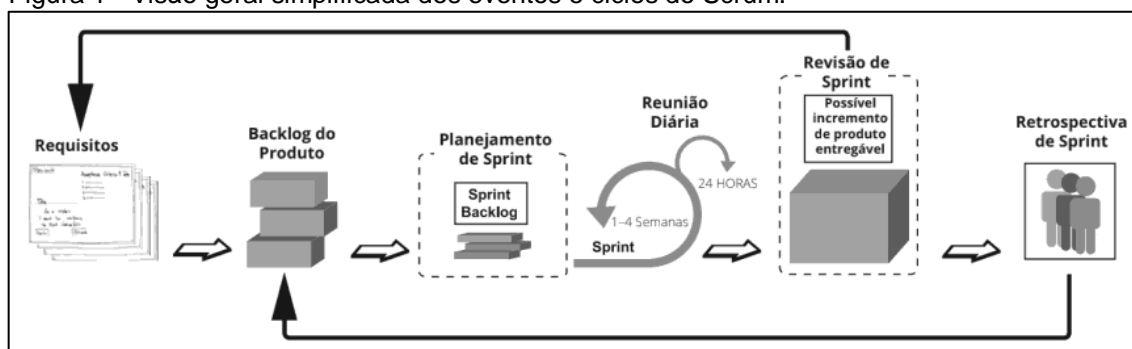
Junior (2017) relata que o MSF é um dos métodos ágeis mais usados por se destinar ao desenvolvimento de soluções tecnológicas por equipes reduzidas, com foco na diminuição de riscos para o negócio e no aumento da qualidade do produto final. O propósito é identificar as falhas mais comuns em projetos de tecnologia, mitigando-as e aumentando as taxas de sucesso de cada iniciativa.

Teixeira (2005) define que *Dynamic Systems Development Methodology* (DSDM) é um dos métodos ágeis mais antigos usados hoje em dia e está inserido no ramo de apoio não só ao desenvolvimento de *software* como no meio tecnológico. O DSDM visa desenvolver uma aplicação com a qualidade desejada sem exceder os limites de tempo e orçamento. Um dos melhores usos está no desenvolvimento incremental e iterativo, a colaboração entre cliente e equipe, além da integração de funcionalidades, testes massificados durante todo o processo e na definição de prioridades entre a lista de requisitos dada pelo cliente. Vale ressaltar que esse *framework* diverge dos demais tanto em sua estrutura, que é composta por processos interligados de modelagem, construção e implementação, como na gestão do tempo, que não é flexível, até permitindo que as funcionalidades mudem, mas desde que os prazos de execução continuem os mesmos.

2.2 Framework Scrum

O *framework* Scrum é uma estrutura organizacional simples para gerenciar projetos complexos. Com característica de ser iterativo e incremental o Scrum não define práticas específicas e detalhada a serem seguidas, é utilizado para tratar e resolver problemas complexos e adaptativos, para desenvolver e entregar produtos com o maior valor possível. É uma ferramenta leve, simples de entender e difícil de dominar pelo fato que não é um processo, técnica ou método. O Scrum é um *framework* que tem a capacidade de ser utilizado em vários processos ou técnicas. Segundo Sabbagh (2013, p. 29), “um *framework* ou arcabouço é uma estrutura básica que pretende servir de suporte e guia para a construção, a partir da expansão dessa própria estrutura, de algo com uso prático”. Na figura 1 apresenta a estrutura organizacional do Scrum mostrando os eventos e ciclos do *framework*.

Figura 1 - Visão geral simplificada dos eventos e ciclos do Scrum.



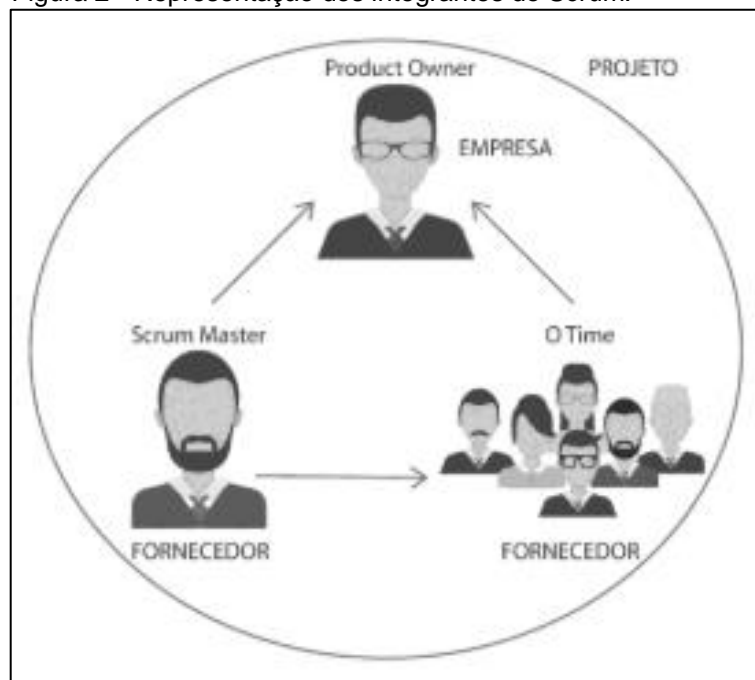
Fonte: Morrow (2019).

Para inicializar o processo de desenvolvimento de um produto com a utilização do *framework* Scrum deve-se começar com uma visão do produto. As características pré-definidas pelo *Product Owner*, as premissas e restrições do produto. Assim, é possível de evitar erros de entendimento e evitando custos de retrabalho. Segundo Sabbagh (2013, p. 39), “a visão do produto é uma forma de traduzir esse objetivo a ser alcançado”.

2.2.1 *Product Owner*

O *Product Owner* é o dono do projeto ou o responsável por representar os *stakeholders*, definindo, comunicando e mantendo constantemente a visão do produto. O *Product Owner* é único e para desempenhar essa função com propriedade, espera que esse profissional tenha conhecimento do negócio ao ponto de saber o que deve ser priorizado para agregar o maior valor possível. Por ser o maior interessado ele deve estar disponível para contribuir com o entendimento e definição da visão do produto, documentos necessários para o início da estrutura do *Product Backlog*. A figura 2 constitui o posicionamento dos integrantes que fazem parte do Scrum. Segundo Silva (2017, p. 13), “ele tem autoridade máxima sobre os itens do *Backlog* do Produto, podendo mudar e incluir itens – é o único que pode fazer isso”.

Figura 2 - Representação dos integrantes do Scrum.



Fonte: Massari (2016).

2.2.2 Scrum *Master*

O Scrum *Master* tem a responsabilidade de promover o Scrum e garantir que quaisquer adversidades encontradas pelo Scrum *Team* sejam solucionadas para que possa estar sempre melhorando a produtividade e a qualidade da equipe, interagindo quando necessário com o Scrum *Team* como um agente de mudança na organização sugerindo novas práticas e se necessário treinando-o para se tornar mais produtivo e a desenvolver produtos de maior qualidade. Segundo Cruz (2013, p. 33) “É o responsável por garantir que o Time Scrum esteja aderindo aos valores do Scrum, as práticas e as regras, ajudando o Time e a organização adotarem o Scrum”.

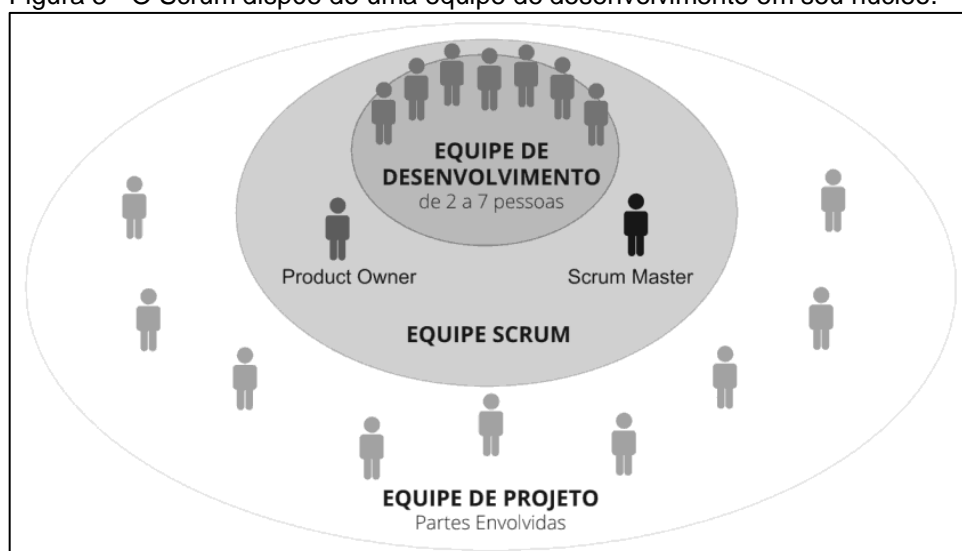
2.2.3 Scrum *Team* ou Time de Desenvolvimento

O Scrum *Team* ou Time de Desenvolvimento é a equipe de profissionais encarregados de desenvolver o projeto. São pessoas auto-organizáveis e multidisciplinares que possuem todos os conhecimentos e habilidades necessárias para gerar o

incremento do produto em cada *Sprint*. Na figura 3 constitui em representar a equipe de desenvolvimento que consiste no Scrum. Segundo Massari (2016, p. 30) “uma equipe auto-organizada é aquela que sabe o que fazer e como deve ser feito, sem depender de uma “voz” de comando para tal”.

O ponto nevrálgico de uma equipe Scrum é a equipe de desenvolvimento – as pessoas que trabalham em estreita colaboração para desenvolver o próprio produto. Elas trabalham diretamente com o *Product Owner* e com o *Scrum Master*, que, por sua vez, alinham as prioridades de negócios e desenvolvimento para a empresa e afastam as distrações, de modo que a equipe de desenvolvimento possa se concentrar no desdobramento de um resultado de qualidade. (MORROW, 2019, p.12).

Figura 3 - O Scrum dispõe de uma equipe de desenvolvimento em seu núcleo.



Fonte: Morrow (2019).

2.2.4 Product Backlog

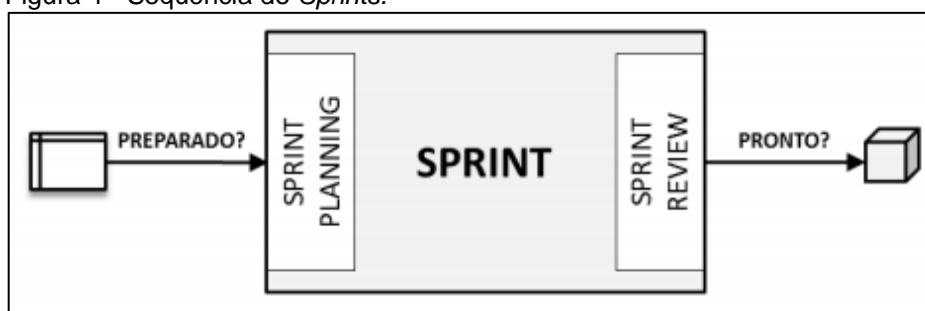
O *Product Backlog* contém as informações que descrevem o produto que será desenvolvido, a classificação ou prioridade para definir o que será realizado primeiro e a estimativa de esforço. Segundo Schwaber (2017, p. 14), “o *Product Backlog* é uma lista ordenada de tudo que é conhecido ser necessário no produto. É a única origem dos requisitos para qualquer mudança a ser feita no produto”.

O *Product Backlog* está em constante evolução e, assim, nunca está terminado ou completo. Conforme o produto evolui, o *Product Backlog* é frequentemente modificado com a adição, subtração, reordenamento e modificação de seus itens. O produto evolui à medida que o ambiente muda e à medida que tanto clientes quanto Time de Desenvolvimento vão conhecendo e entendendo melhor esse produto que está sendo construído. (SABBAGH, 2013, p. 112).

2.2.5 Sprint Planning

A *Sprint Planning* é uma reunião que contém a presença obrigatória do *Product Owner*, *Scrum Master* e o *Scrum Team* para que possam negociar as metas de desenvolvimento da *Sprint*, gerando uma lista de itens do *Product Backlog* para ser desenvolvida e somando com o plano de trabalho que será realizado, sendo chamado de *Sprint Backlog*. Na figura 4 apresenta a sequência de eventos dentro da *Sprint*. Segundo Sabbagh (2013, p. 223) “todos os membros do Time de Desenvolvimento participam da reunião com igual poder de opinião e decisão, quaisquer que sejam suas áreas de conhecimento ou de atuação”.

Figura 4 - Sequência de Sprints.



Fonte: Sabbagh (2013).

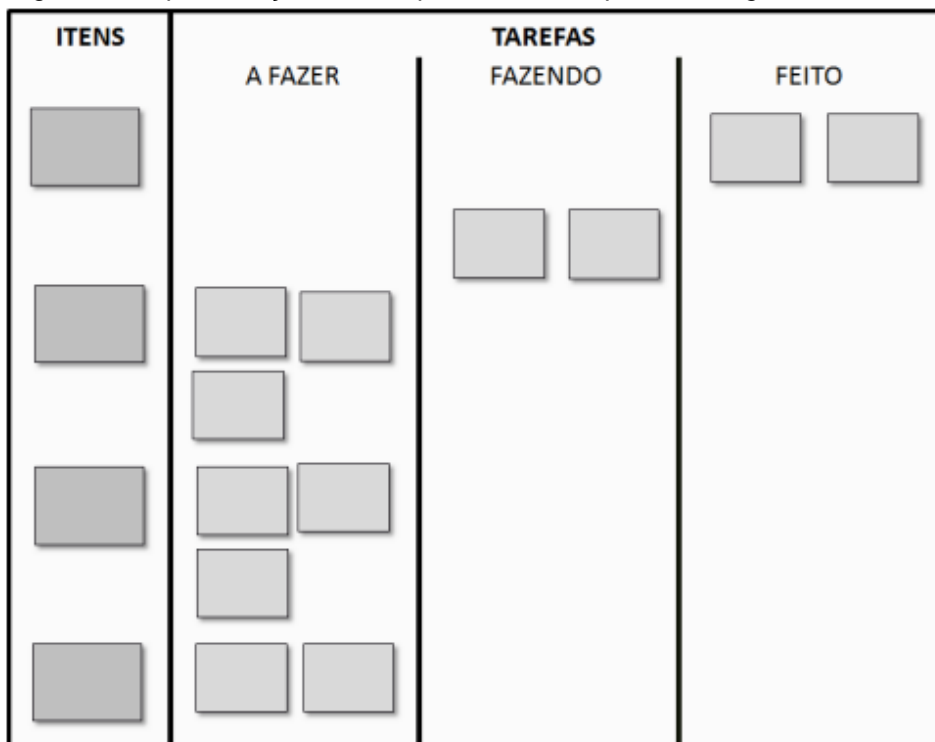
2.2.6 Sprint Backlog

O *Sprint Backlog* é a realização do plano que define a quantidade de itens do *Product Backlog* que serão feitos pelo *Scrum Team* no período da *Sprint*. Na figura 5 apresenta um quadro que consta um conjunto de tarefas que correspondem a cada item, indicando o andamento de cada tarefa, assim, podendo informar se a tarefa não foi inicializada, se está em andamento ou já está concluída. É possível adicionar estimativas de tempo de desenvolvimento para cada tarefa para que o *Scrum Team* possa acompanhar o progresso da conclusão da *Sprint*. Segundo Silva (2017, p. 18), “o *Backlog* da *Sprint* sempre terá uma meta a ser atingida, a ser determinada na reunião de planejamento da *Sprint* após a seleção dos itens conforme a prioridade determinada pelo Dono do Produto”.

O *Sprint Backlog* é, portanto, é um conjunto de itens do *Product Backlog* selecionados para a *Sprint*, juntamente com o plano para entregar o incremento do produto e atingir o objetivo da *Sprint*. O *Backlog* da *Sprint* é a previsão do Time de Desenvolvimento sobre qual funcionalidade estará no próximo

incremento e sobre o trabalho necessário para entregar essa funcionalidade em um incremento Pronto. (SCHWABER; SUTHERLAND, 2017, p. 16).

Figura 5 - Representação em um quadro real do *Sprint Backlog*.

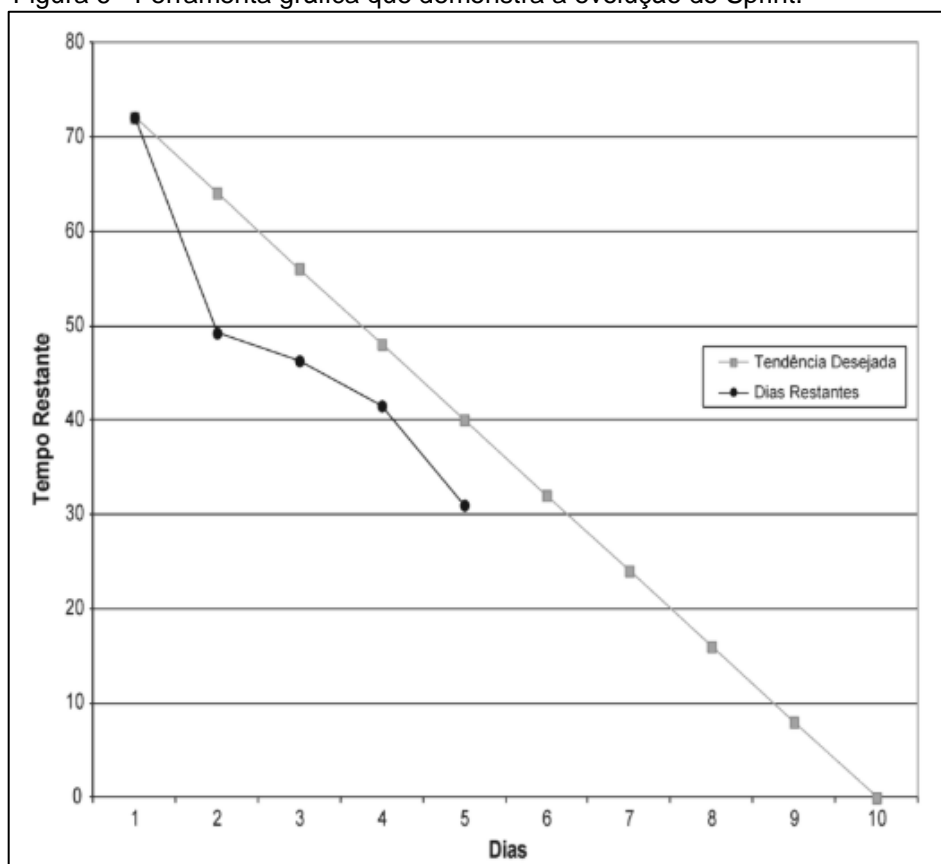


Fonte: Sabbagh (2013).

2.2.7 Sprint

No Scrum uma iteração ou ciclo de trabalho é definido como *Sprint*, ou seja, é um *time boxing* que dedica um determinado período de tempo fixo dentro do qual a atividade planejada ocorre. E um *Sprint* novo só pode ser inicializando quando o anterior estiver concluído. Com isso, dedica-se a concluir os itens da *Sprint* estabelecidos no *Sprint Planning*. Na qual se planeja as atividades a serem realizadas no próprio *Sprint*. Com o uso da ferramenta gráfica, o *Sprint Burndown* é demonstrado na figura 6 apresentando de forma ilustrativa a evolução da *Sprint* e sendo capaz de visualizar o que falta para ser executado nos itens selecionados para a *Sprint*. Segundo Silva (2017, p.18) “durante a execução da *Sprint*, uma vez por dia, o trabalho que ainda resta deve ser reportado de forma que possa obter o *Sprint Burndown*”.

Figura 6 - Ferramenta gráfica que demonstra a evolução do Sprint.



Fonte: Silva (2017).

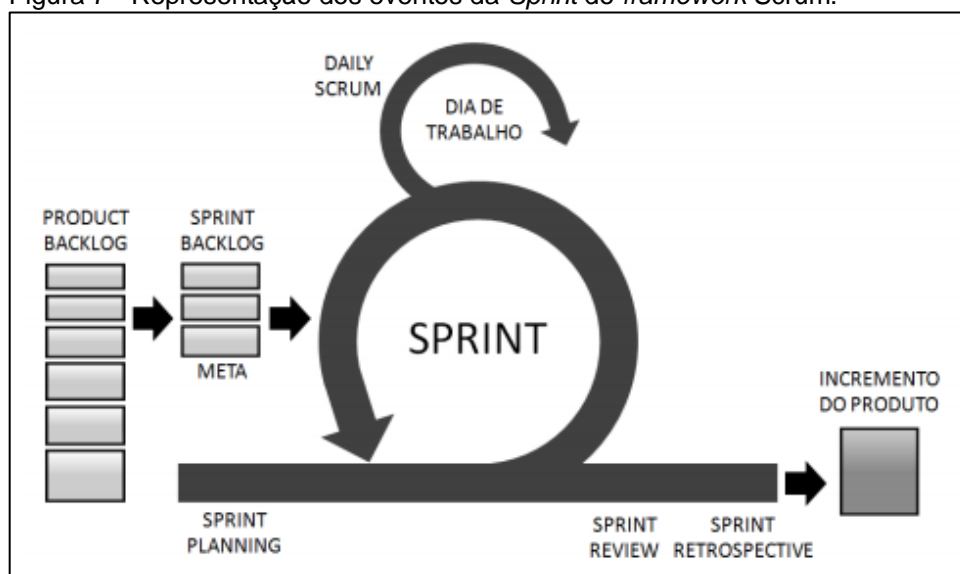
2.2.8 Daily Scrum

O *Daily Scrum* ou Reunião Diária do Scrum também é um evento *time boxing* de aproximadamente 15 minutos para que o Scrum *Team* possa planejar os trabalhos diários, melhorando a colaboração e a auto-organização da equipe através da inspeção do desenvolvimento do trabalho desde a última reunião. Segundo Cruz (2013, p. 37) “O foco das perguntas é um alinhamento do que foi realizado e do que será realizado, que poderá agregar valor aos trabalhos dos outros membros. A reunião não deve ser considerada ou usada como uma reunião de passagem de status”.

2.2.9 Sprint Review

A *Sprint Review* ou Revisão da *Sprint* é realizada após a conclusão da *Sprint* para verificar o incremento e quando necessário adaptar o *Product Backlog*. A presença do *Scrum Team*, do *Product Owner* e do *Scrum Master* são obrigatórias nessa reunião. Esta é uma reunião informal com o objetivo de obter o *feedback* do cliente e dos demais interessadas sobre o incremento do produto desenvolvido até o momento, portanto, essa reunião é para inspecionar e adaptar (quando necessário) o produto, assim, o *Product Owner* poderá utilizar como matéria-prima para modificar o *Product Backlog* para *Sprints* futuros. Segundo Cruz (2013, p. 37) “Será possível conferir e avaliar o que está sendo considerado pronto, levando em conta o que está sendo entregue *versus* o que deveria ser entregue”. Na figura 7 constitui em mostrar a visão geral dos eventos realizados durante a *Sprint*.

Figura 7 - Representação dos eventos da *Sprint* do framework Scrum.



Fonte: Sabbagh (2013).

2.2.10 Sprint Retrospective

A *Sprint Retrospective* é o momento onde o *Scrum Team* inspeciona a si próprio e avaliam a *Sprint* passada, levando em conta também as pessoas, relacionamento, processos e as ferramentas utilizadas, assim, criando um plano de melhorias a ser aplicada no próximo *Sprint*. Ela ocorre logo em seguida da *Sprint Review* e es-

se momento é utilizado para melhorar a forma como as coisas devem ser feitas. O *Scrum Master* tem uma grande importância pelo fato que nesse evento precisa ser positivo, identificando os principais itens que foram bem e as potenciais melhorias. Segundo Cruz (2013, p. 38) “No final da reunião de retrospectiva, o Time deve sair com uma identificação de medidas de melhoria factíveis que serão implementadas na próxima *Sprint*”.

2.3 Definições

Schwaber (2017) define que O *framework* Scrum é definido como uma ferramenta para a gestão dinâmica de projetos, sendo muitas vezes aplicadas para o desenvolvimento ágil para que as pessoas possam tratar e resolver problemas complexos e adaptativos, enquanto estão produzindo e criando produtos com o mais alto valor possível, ou seja, é uma ferramenta que permite controlar de forma eficaz e eficiente o trabalho, potencializando as equipes que trabalham em prol de um objetivo em comum. Sendo leve, simples de entender e difícil de dominar o Scrum está sendo usado para gerenciar o trabalho de produtos complexos desde o início da década de 90. O *framework* não se define como um processo, técnica ou um método definitivo. Ao contrário, é capaz de empregar vários processos ou técnicas de trabalho, de modo que possa continuamente melhorar o produto, o time e o ambiente de trabalho. No *framework* Scrum é composto por times Scrum associados a papéis, eventos, artefatos e regras. Cada componente ou integrante dentro do *framework* possui um propósito específico e é essencial para o uso e sucesso do Scrum. As regras do Scrum integram os papéis, eventos e artefatos apresentados na tabela 1, administrando as relações e interações entre eles. Prescrevendo apenas três papéis *Product Owner*, Time de Desenvolvimento e *Scrum Master*, eles são utilizados popularmente nas divisões de hierarquias das empresas.

Tabela 1 - Apresenta as práticas do Scrum.

PAPÉIS	EVENTOS	ARTEFATOS
<i>Product Owner</i>	Reunião de planejamento da <i>Sprint</i>	<i>Product Backlog</i>
<i>Scrum Master</i>	Reunião diária	<i>Sprint</i>
<i>Scrum Team</i>	Reunião da revisão da <i>Sprint</i>	Produto ou funcionalidade concluída

Fonte: DEVMEDIA (2016).

A base fundamental do Scrum é representada na figura 8:

Figura 8 - A base fundamental do *framework* Scrum.



Fonte: *Mind master* (2015).

Dinamize (2020) define que quando é tomada a decisão de realizar a prática do Scrum em um projeto ou produto, uma das primeiras coisas que se deve compreender é que há times multifuncionais que possuem todas as competências para completar o trabalho sem depender de outros que não fazem parte da equipe, ou seja, há necessidade de entender que as funções e papéis no *framework* Scrum são diferentes das funções tradicionais na execução de um projeto. Como visto anteriormente, apesar de existir apenas três papéis fundamentais é necessário também compreender como é dividido cada termo utilizado no *framework*, na tabela 2 é possível visualizar o significado de cada termo utilizado no Scrum:

Tabela 2 - Termos utilizados no *Framework Scrum*.

TERMO	SIGNIFICADO
PRODUCT OWNER	É o responsável por definir os itens que irão compor o <i>product backlog</i> e por dirigir o <i>Sprint planning meeting</i> .
SCRUM TEAM	Equipe de desenvolvimento que realizam o trabalho.
SCRUM MASTER	Responsável por promover e suportar que as melhores práticas do <i>framework</i> estão sendo aplicadas.
PRODUCT BACKLOG	Documento com todos os requisitos a serem desenvolvidos de acordo com as prioridades e com as funcionalidades a serem entregues aos clientes.
SPRINT	Tempo em que o projeto deve ser realizado.
SPRINT PLANNING	Reuniões para definir quais atividades serão desenvolvidas em determinado <i>Sprint</i> .
DAILY SCRUM	Reunião diária para definir as tarefas do dia.
SPRINT REVIEW	Inspecionar o incremento e adaptar o <i>Product Backlog</i> se necessário.
SPRINT RETROSPECTIVE	Reunião para revisar pontos positivos e negativos do <i>Sprint</i> que foi realizado.
SPRINT BACKLOG	É uma lista de tarefas que o <i>Scrum Team</i> se compromete a fazer em um <i>Sprint</i> .
SPRINT REVIEW MEETING	Reunião na qual todo trabalho feito é revisado. Apresenta as demos para os clientes.
SPRINT BURNDOWN	Ferramenta gráfica que facilita o acompanhamento dos <i>Sprints</i> .
INCREMENTO	É a soma de todos os itens do <i>Product Backlog</i> completados durante a <i>Sprint</i> e o valor dos incrementos de todas as <i>Sprints</i> anteriores
“PRONTO”	Quando um item do <i>Product Backlog</i> ou um incremento é descrito como “pronto”.

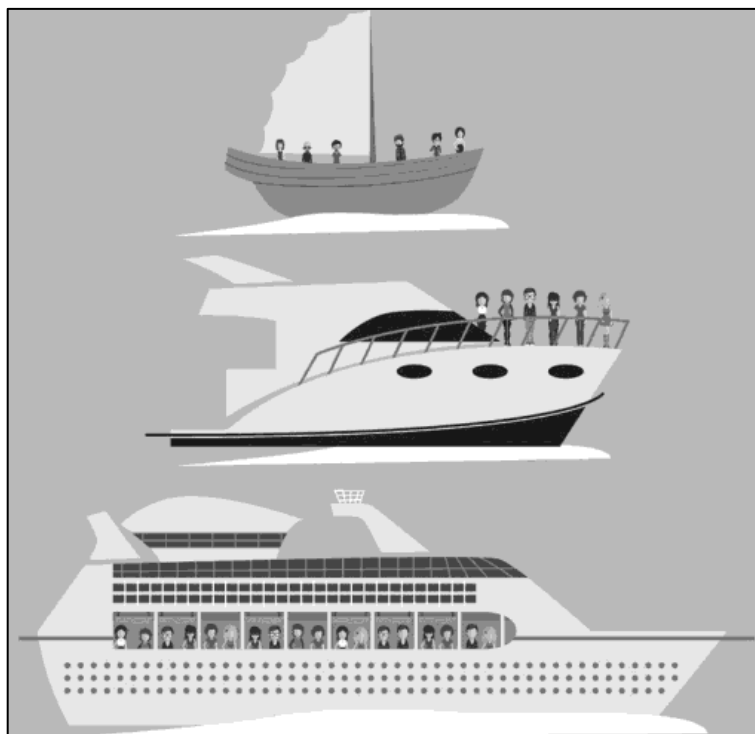
Fonte: O autor baseado em Schwaber (2017).

2.3.1 MVP (*Minimum Viable Product*)

Outra definição muito importante do Scrum é possibilitar a criação de um *Minimum Viable Product* ou Mínimo Produto Viável (MVP), também conhecido como *Minimally Marketable Feature* ou Funcionalidade Mínima Comercializável (MMF), que significa o conjunto mínimo de funcionalidades de um produto. Essa definição tem como ideia de lançar um novo produto ou solução com as mínimas funcionalidades necessárias com o menor investimento possível. No segmento automotivo seriam desenvolvidos protótipos para que possa ser testado e avaliado a viabilidade das funcionalidades técnicas do produto. Também será possível observar a usabilidade do veículo, a aceitação do mercado, comparação entre os concorrentes e se atenderá os *stakeholders*. Massari (2016, p. 129) relata que “nem sempre o MVP é um produto já disponível para o cliente, e sim uma prévia do produto sobre o qual o cliente pode dar *feedback*, solicitar mudanças ou mesmo abortar o projeto”.

Manchada (2019) define que a realização do MVP faz parte de um processo de evolução do produto, e construir implica em encontrar o equilíbrio certo entre o que empresa desenvolvedora está oferecendo aos usuários e o que os consumidores realmente desejam. O objetivo do MVP é analisar a hipótese minimizando erros, auxiliando no levantamento de informações, visando identificar grupos ou tipos específicos de usuários. Na figura 9 vemos um exemplo de MVP, que tem o objetivo inicial de atravessar um rio em segurança com algumas pessoas a bordo:

Figura 9 - Exemplo de evolução de um MVP.



Fonte: Massari (2016).

Bianchi (2019) define que com base na própria terminologia, o MVP tem como objetivo de apresentar uma versão reduzida e simplificada do produto que se pretende desenvolver. É importante lembrar que é necessário possuir no mínimo todas as funções que o produto final deverá ter, para que as análises sejam confiáveis. A utilização desta metodologia permite que, com o auxílio das respostas dos usuários e da forma como os concorrentes reagem seja possível entender o comportamento do produto ou serviço no mercado consumidor. Assim, possibilitando que a organização entenda se a ideia é factível e lucrativa antes mesmo de realizar grandes investimentos. E se o retorno for positivo a empresa também consegue observar qual é a diferença entre os valores reais e os valores previstos para o serviço, possibilitando o aprimoramento da ideia e garantindo maior aceitação e maiores margens de lucro.

2.3.2 Stakeholders

Bezerra (2014) define que o termo *stakeholders* pode ser uma pessoa ou grupo que tem participação, investimentos ou ações e possui interesse em um determinado negócio ou empresa, ou seja, refere-se às partes interessadas ou público es-

tratégico que definem quais são os objetivos e as necessidades do projeto. No contexto do termo oriundo do inglês, a palavra *stake* significa interesse, participação e risco. Enquanto *holder* significa aquele que possui. É uma expressão utilizada em várias áreas como gestão de projetos, arquitetura de *software* e na área de tecnologia da informação. Assim, descrevendo uma pessoa ou grupo que possuem interesse em determinado negócio, indústria ou serviço. Com exemplos de *stakeholder* associamos apenas no segmento de projetos, e por isso, seria o gerente, patrocinador, a equipe de desenvolvimento, o cliente e o colaborador no qual solicita o projeto e tem a responsabilidade de contribuir com recursos financeiros, mas na realidade podem ser várias outras partes interessadas, podendo ser o concorrente, fornecedor e o investidor. Deste modo, o termo *stakeholder* tem se tornado cada vez mais comum, Na figura 10 ilustra uma peça muito importante para contribuir no desempenho de uma determinada organização e influenciar nas atitudes e ações dentro da empresa, ou seja, as partes interessadas devem estar de acordo com as práticas de governança corporativa executadas pela empresa.

Figura 10 – Ilustração de *Stakeholders*.

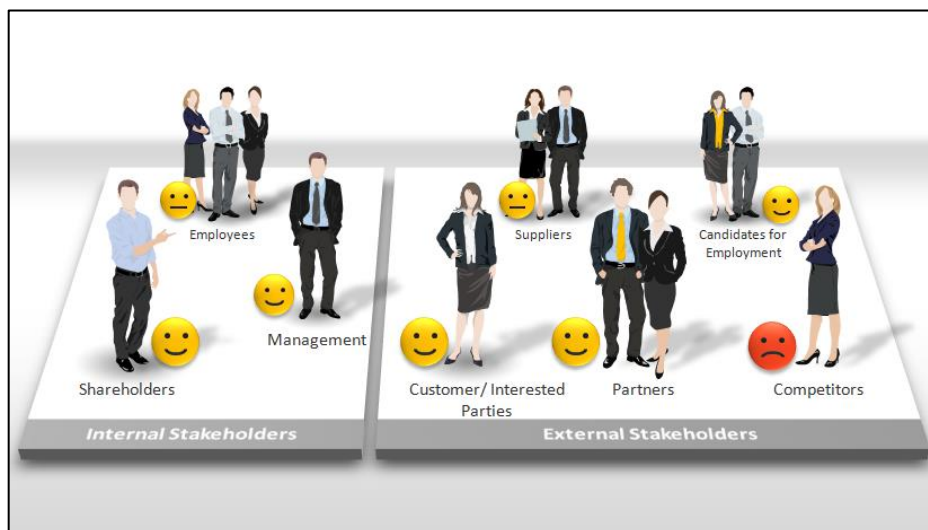


Fonte: Bezerra (2014).

Trindade (2011) define que a função dos *stakeholders* tem a capacidade de afetar diretamente a organização e por isso não pode ser menosprezado. Por isso, hoje em dia temos *stakeholders* que estão prontos para penalizar as empresas em defesa de uma determinada causa e o papel deles tem afetado as organizações e,

além disso, os passos que são dados nos negócios levam isso em consideração. Na figura 11 apresenta diversos tipos de *stakeholders*.

Figura 11 - Orientação das partes interessadas (*stakeholders*).



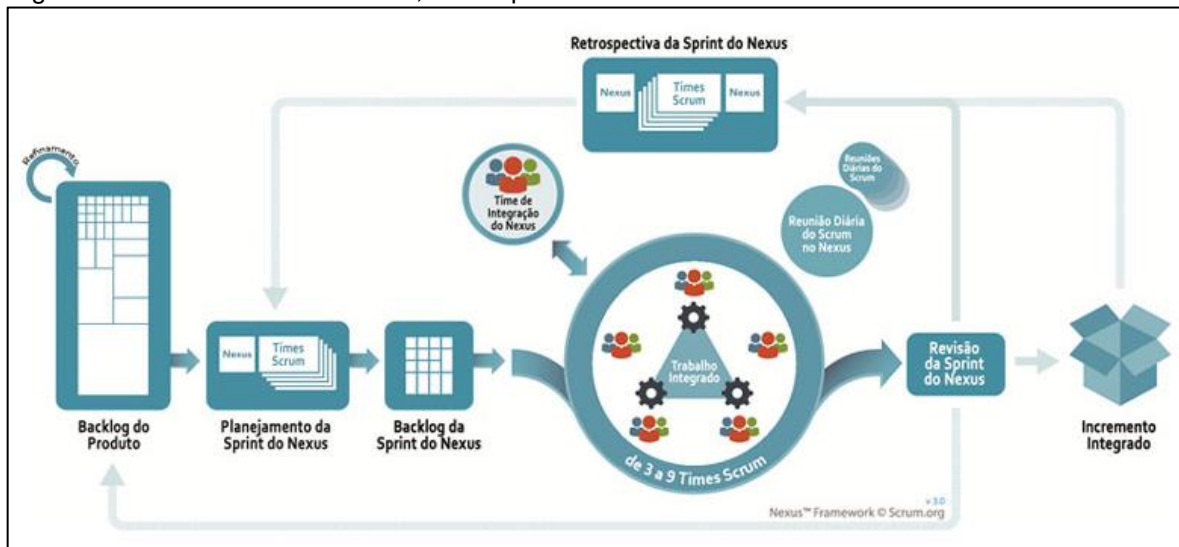
Fonte: *Mayr's Organization Management* (2013).

2.4 Nexus framework Scrum para escalonamento ágil

Kendis (2018) define que o Nexus é um *framework* que implementa o Scrum como alicerce para sua construção em escala para fornecer um único produto integrado, ou seja, para o desenvolvimento e sustentação de iniciativas de criação de produtos ou de *softwares* em escala. Podendo ser utilizado entre 3 e 9 Scrum *Teams* que estejam trabalhando em um mesmo *Product Backlog* e estando dedicados em produzir um incremento combinado a cada *Sprint* com dependências mínimas. Segundo Schwaber (2015, p. 4) “Nexus é um exoesqueleto que permeia sobre múltiplos Times Scrum quando estes estão organizados para criar um Incremento Integrado. O Nexus é consistente com o Scrum e suas partes serão familiares para aqueles que trabalharam com projetos Scrum.”.

Schwaber (2015) define que a diferença entre o *framework* Scrum e o *framework* Nexus é o foco dado para as dependências e de se comunicar de forma transparente com outro Scrum *Team* durante o desenvolvimento, entregando um incremento combinado e “pronto” pelo menos a cada *Sprint*. Na figura 12 mostra a estrutura denominada de exoesqueleto do Scrum Escalado:

Figura 12 - O Framework Nexus™, exoesqueleto do Scrum Escalado.



Fonte: Guia do Nexus™ (2015).

Pereira (2018) define que o Time de Integração do Nexus é constituído pelo *Product Owner*, *Scrum Master* e os membros do Time de Integração do Nexus. E são responsáveis por garantir que ao menos seja realizado um incremento devidamente integrado e “pronto” em cada *Sprint*. Segundo Schwaber (2015, p. 4) “Um novo papel, o Time de Integração do Nexus, este existe para coordenar, treinar e supervisionar a aplicação do Nexus e a operação do Scrum para que os melhores resultados sejam obtidos”. Na figura 13 apresenta o Time de Integração que é muito parecido ao do Scrum, mas possuem suas particularidades. Sendo composto de:

O *Product Owner* é semelhante ao do Scrum, mas no Nexus há apenas um *Product Backlog*. O *Product Owner* faz parte do time de integração e sendo o responsável pelo *Product Backlog*, incluindo seu conteúdo, disponibilidade e ordenação.

O *Scrum Master* no Time de Integração do Nexus também é semelhante ao do Scrum, possuindo as mesmas responsabilidades e garantindo que o *framework Nexus* seja entendido e aplicado.

O Time de Integração do Nexus normalmente é formado por profissionais de *software* que são qualificados no uso da prática de integração. Eles são os responsáveis por garantir que essas práticas e ferramentas sejam integradas na aplicação. E também são responsáveis por garantir que essas práticas e ferramentas sejam

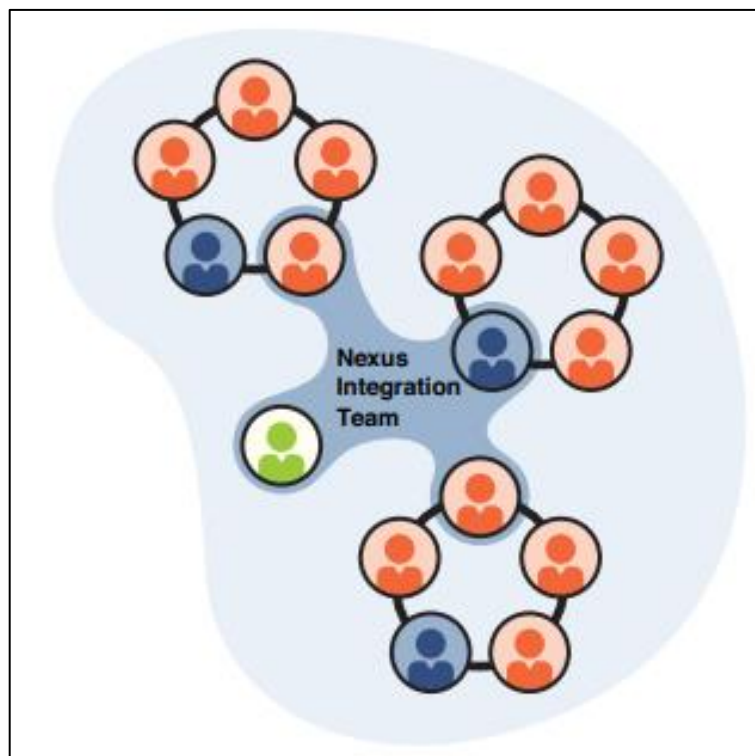
implementadas, entendidas e usadas. Na figura 14 apresenta a estrutura utilizada pelo Time de Integração.

Figura 13 - Membros da equipe de integração do Nexus.



Fonte: Bittner (2017).

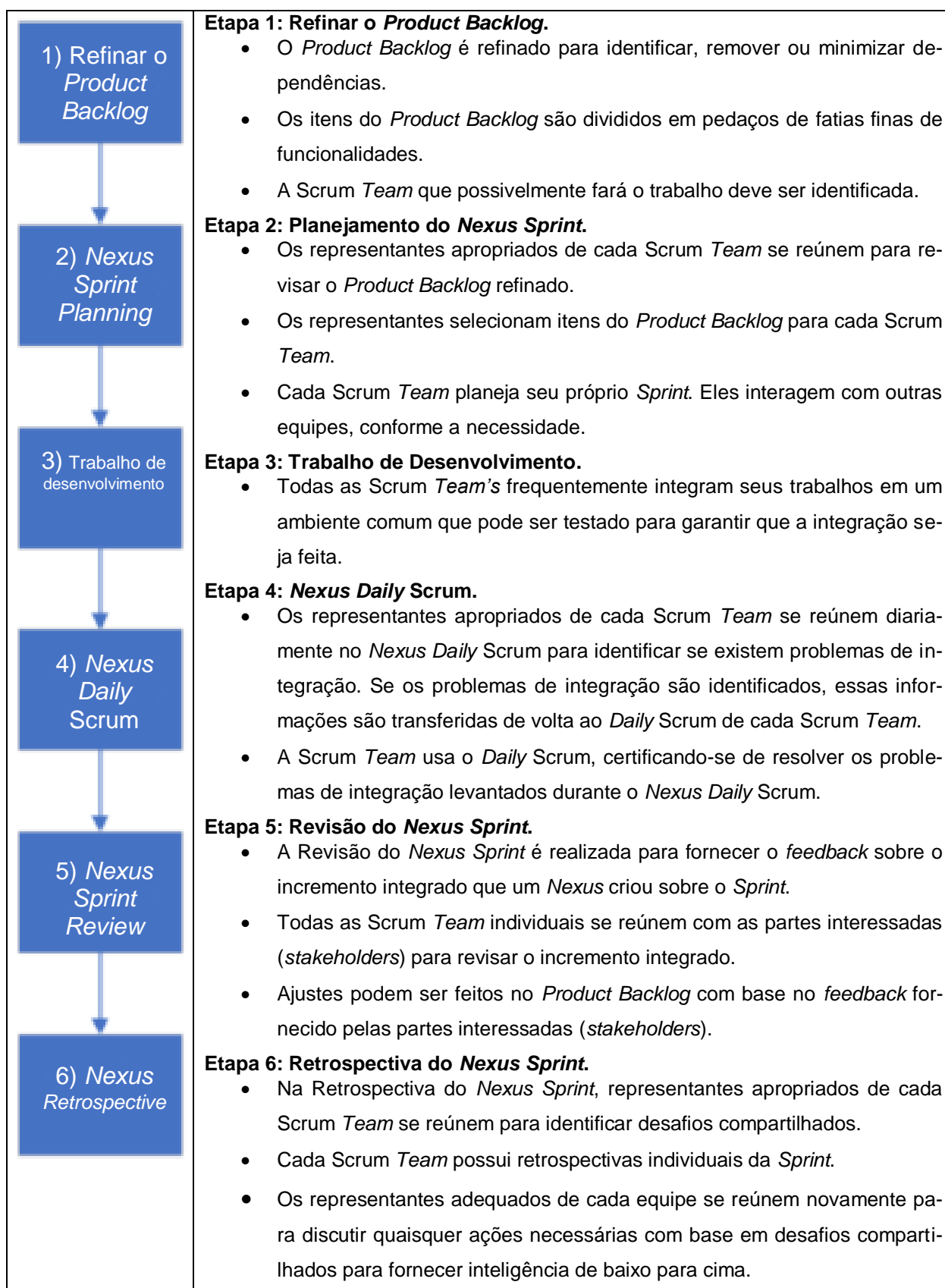
Figura 14 - Estrutura do Time de integração do Nexus.



Fonte: Bittner (2017).

2.4.1 Fluxo de processo do *framework Nexus*

Schwaber (2015) define que o todo o trabalho realizado no *framework Nexus* é possível ser feito por quaisquer membros do *Scrum Team* visto que todos os membros são multifuncionais do *Nexus*. Com isso, é possível basear nas dependências, os *Scrum Team's* podem escolher os membros mais qualificados para fazerem um determinado trabalho específico. Na tabela 3 informa como o fluxo de processo do *framework Nexus* funciona.

Tabela 3 - Fluxo de processo do *Nexus framework* Scrum.

Fonte: Bathia (2019).

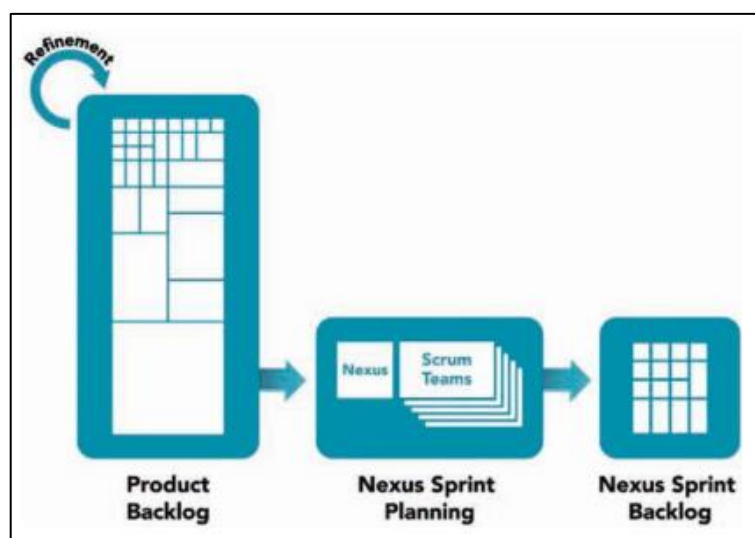
2.5 Os eventos do *framework Nexus*

Pereira (2018) define que os eventos do *framework Nexus* são baseados no Scrum e os *times boxes* de todos os eventos são os mesmos. Segundo Schwaber (2015, p. 8) “A duração dos eventos do *Nexus* é guiada pelos tamanhos dos seus eventos correspondentes no Guia do Scrum. Além de serem *times boxes* de seus eventos correspondentes no Scrum.”.

2.5.1 A Reunião de Planejamento do *Nexus*

Pereira (2018) define que a reunião de planejamento do *Nexus* tem o propósito de realizar o alinhamento entre as equipes de desenvolvimento para organizar as atividades de todos os *Scrum Team* para uma única *Sprint*. Na reunião de planejamento do *Nexus* é formulada a meta da *Sprint Nexus*, descrevendo o objetivo a ser alcançado pelos *Scrum Teams* durante a *Sprint*. Em seguida, as equipes de desenvolvimento podem realizar individualmente seus eventos de *Sprint Planning*. O *Nexus* usa o *Product Backlog* refinado como entrada para o evento *Nexus Sprint Planning*. Segundo Schwaber (2015, p. 8) “Novas dependências podem emergir durante o Planejamento da *Sprint* do *Nexus*. Estas devem ser visualizadas e minimizadas. A sequência do trabalho entre os *times* pode também ser ajustada. Um refinamento adequado do *Backlog* do Produto minimizará o surgimento de novas dependências durante o Planejamento da *Sprint* do *Nexus*.”. Na figura 15 representa a sequência básica para realizar o *Nexus Sprint Planning*.

Figura 15 – Sequência para realização da *Nexus Sprint Planning*.



Fonte: Bittner (2017).

2.5.2 Reunião Diária do Scrum no *Nexus*

Schwaber (2015) define que a Reunião Diária do Scrum no *Nexus* é um evento dedicado para os representantes das *Scrum Teams* analisarem individualmente a situação real em que está o Incremento Integrado e detectar erros ou problemas de integração e novas descobertas sobre as dependências entre os times. Segundo Team (2018, p. 1) “O objetivo desta reunião é coordenar quaisquer desafios e dependências do dia em que todas as equipes estejam cientes”.

2.5.3 Revisão da *Sprint* do *Nexus*

Kendis (2018) define que a Revisão do *Nexus Sprint* é feita no fim da *Sprint* e reúne todas as equipes de desenvolvimento do Scrum junto ao *Product Owner* para revisar e proporcionar o retorno do incremento integrado que o *Nexus* desenvolveu. Segundo Schwaber (2015, p. 9) “A Revisão da *Sprint* do *Nexus* substitui as Reuniões individuais de Revisão do Scrum, porque todo o incremento integrado é o foco para capturar o retorno das partes interessadas. Pode não ser possível mostrar todo o trabalho completado em detalhe”.

2.5.4 Retrospectiva da *Sprint* do *Nexus*

Schwaber (2015) define que a Retrospectiva da *Sprint* do *Nexus* é o momento ideal para o *Nexus* dedicar na inspeção e adaptação. Sendo dividido em três partes:

- Na primeira parte é o momento ideal para que os representantes de todo o *Nexus* possam entender e identificar os problemas que impactam em uma de um time. O objetivo é fazer com que todos os problemas apresentados possam ser solucionados e compartilhados para todos os *Scrum Teams*.
- Na segunda parte todos os *Scrum Teams* fazem suas próprias *Sprint Retrospective* como é descrito no *framework* Scrum. Os problemas apresentados na Retrospectiva da *Sprint* do *Nexus* podem ser usados como entrada das discussões do time.
- Na terceira e última parte visa em reunir novamente os representantes de todos os *Scrum Teams* para chegarem a um acordo sobre como monitorar e controlar as ações identificadas.

Kendis (2018) define que Retrospectiva do *Nexus Sprint* é a essência de toda retrospectiva para atender e identificar os desafios compartilhados por todos os representantes do *Nexus*. As soluções são apresentadas e discutidas com objetivo de analisar como as ideias podem auxiliar na solução dos problemas. Em seguida, a equipe *Nexus* e as *Scrum Teams* possuem suas respectivas retrospectivas. Por fim, há uma retrospectiva coletiva em que as soluções são compartilhadas com todo o *Nexus* e as *Scrum Teams*.

2.5.5 Refinamento

Schwaber (2015) define que no *Nexus* possui diversos níveis de refinamento, isso acontece quando os itens do *Product Backlog* são independentes e podem ser realizados sem grandes conflitos entre os *Scrum Teams* do *Nexus*. Esse refinamento do *Product Backlog* em escala faz com que sirva para dois propósitos. Auxiliar a prever qual time irá entregar quais itens do *Product Backlog* e detecta dependências entre os times, assim, sendo possível permitir aos times monitor e minimizar as dependências entre os times. Segundo Kendis (2018, p. 1) “Os itens da lista de dependências do produto são refinados continuamente para minimizar ou limpar dependên-

cias. Novos requisitos também podem ser adicionados. Também inclui a realização de estimativas relevantes dos pontos da história”.

2.6 Os artefatos do *framework Nexus*

Kendis (2018) define que os artefatos para o *Nexus framework* são que todos os *Scrum Teams* utilizam o mesmo *Product Backlog*, os *Scrum Teams* mantem seu *Backlog* da *Sprint* individual e o *Nexus Sprint Backlog* é a coleção de *Backlogs* individuais do *Sprint* que serve para auxiliar a visualizar com transparência e destacar as dependências durante a *Sprint*. Segundo Schwaber (2015, p. 11) “Artefatos representam o trabalho ou valor que fornece transparência e oportunidades para inspeção e adaptação, assim como descrito no Guia do Scrum.”.

Bittner (2018) define que no Scrum, o refinamento do *Product Backlog* não é um evento obrigatório na realização do *framework Nexus*, mas é uma boa prática. O refinamento é importante para ajudar as *Scrum Teams* trabalharem juntas para definir quem irá realizar os itens do *Product Backlog* e visualizar as dependências entre equipes.

2.6.1 *Backlog* do Produto

Bittner (2018) define que existe apenas um único *Product Backlog* para todo o *framework Nexus* e para todos os *Scrum Teams*. O *Product Owner* é o responsável por organizar um único *Product Backlog* e todas as equipes retiram atividades a serem realizadas deste único artefato. Segundo Schwaber (2015, p. 11) “Os Itens do *Backlog* do Produto são considerados “preparados” para a Reunião de Planejamento do *Nexus* quando podem ser selecionados para serem feitos pelos Times Scrum sem ou com uma dependência mínima de outros Times Scrum”.

2.6.2 Meta do *Nexus*

Bittner (2018) define que a meta do *Nexus* é estabelecida pelo *Product Owner* durante a reunião de planejamento da *Sprint*, este artefato tem a função de somar todos os trabalhos e metas das *Sprints* individuais de cada *Scrum Team* do *Nexus* e

consequentemente o *Nexus* precisa demonstrar a funcionalidade desenvolvida para alcançar a meta do *Nexus* na revisão do *Nexus Sprint*.

2.6.3 Backlog da Sprint do Nexus

Bittner (2018) define que o *Backlog* do *Nexus Sprint* possui todos os Itens do *Product Backlog* (PBI) que tenham dependências entre os *Scrum Teams*, seu objetivo é ser utilizado para ressaltar as dependências e o fluxo de trabalhando durante o *Sprint*. Este é atualizado diariamente durante o evento da reunião diária do *Nexus*.

2.6.4 Incremento Integrado

Bittner (2018) define que um incremento integrado é a junção de todos os trabalhos realizados por todos os *Scrum Teams* do *Nexus*, o incremento integrado deve ser utilizável e atender a realidade de entrega, significa que este deve atender os conceitos de “pronto”. O *Product Owner* é um dos principais interessados (*stakeholder*) e determina os critérios de qualidade que o incremento do produto deva atender. Este artefato é inspecionado na revisão do *Nexus Sprint*.

2.7 Modelo Cascata (*framework Waterfall*)

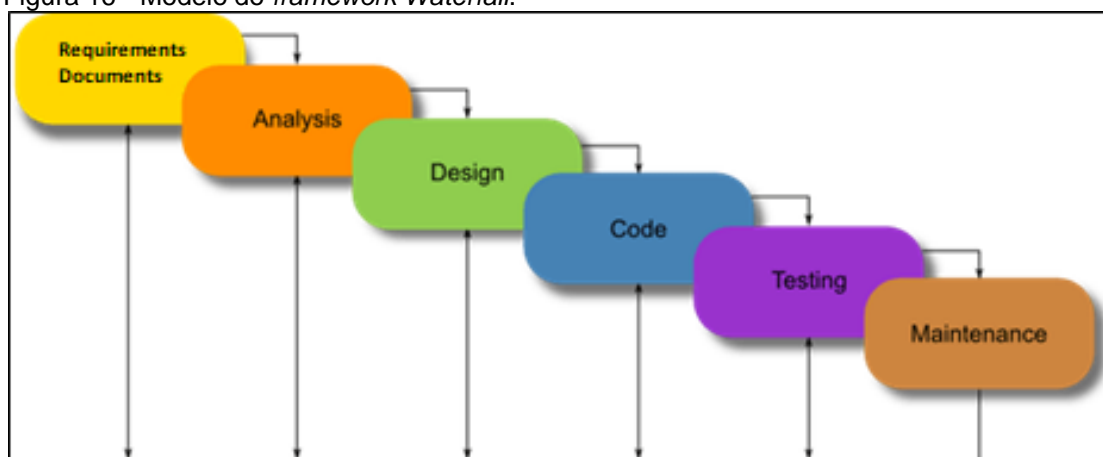
Morse (2016) define que o *framework Waterfall* ou Modelo em Cascata foi apresentado pela primeira vez em um artigo no ano de 1970 pelo Dr. Winston W. Royce, o Modelo Cascata é um processo de desenvolvimento de *software* que visa à progressão lógica de etapas que sejam executadas ao longo do ciclo de desenvolvimento do *software*, ou seja, sendo uma cascata incremental. A implementação desse modelo em um projeto de *software* é considerado simples graças ao fato que grande parte da natureza do passo a passo do próprio Modelo em Cascata.

Campos (2018) define que o Modelo Cascata é uma espécie de roteiro que determina a realização de atividades de forma sequencial no período de desenvolvimento de um projeto. O principal objetivo é garantir que todas as ideias possam ser analisadas sob todos os aspectos para que nada fique de fora.

Lotz (2018) define que em um verdadeiro projeto de desenvolvimento do *framework Waterfall*, cada item representa um estágio do processo de desenvolvimento do *software*, e normalmente cada estágio é encerrado quando realmente é terminado para que possa iniciar a nova etapa. A figura 16 representa o modelo linear dos estágios de cada processo de desenvolvimento do Modelo Cascata (*framework Waterfall*).

Morse (2016) define que o Modelo Cascata é uma abordagem linear para o desenvolvimento de *software*. Neste *framework Waterfall*, a sequência de eventos é algo como:

- **Requisitos:** Nesta fase inicial, tem como objetivo coletar os requisitos e informações necessárias sobre o projeto para gerar um documento chamado *Requirements Document* que registra as definições do projeto que devem ser realizadas de forma clara.
- **Análise:** Nesta fase, a equipe de desenvolvimento realiza a análise para estabelecer o modelo de programação que será utilizado no projeto. Ou seja, resultando no desenho da arquitetura geral do sistema com base nas informações adquiridas na fase anterior.
- **Design:** A codificação não é realizada nesta fase, mas esta fase abrange os requisitos técnicos de linguagem de programação e *hardware*.
- **Codificação:** Nesta fase a codificação é realizada. Englobando os requisitos técnicos, modelos, linguagem de programação e especificações.
- **Teste:** Durante a fase de Teste os desenvolvedores realizam diversos testes para encontrar e relatar quaisquer erros ou problemas. Não é anormal que esta fase faça com que retorne a fase de Codificação para que os erros encontrados sejam devidamente corrigidos.
- **Operações:** Por fim, o *software* está concluído e pode ser implementado ou liberado em um ambiente ativo. Na medida em que surge a necessidade de correção de erros ou melhorias a equipe de desenvolvimento está disponível para suporte, manutenção e se necessário em criar atualizações.

Figura 16 - Modelo do *framework Waterfall*.

Fonte: O autor baseado em Morse (2016).

2.7.1 As vantagens do uso do Modelo Cascata (*framework Waterfall*)

Dias (2019) define que o Modelo Cascata por natureza é separado em unidades, e possibilita implementação simultânea de vários programadores, aumentando a agilidade de entrega. Outra grande vantagem é quando o escopo do trabalho é claramente definido. E se todas as especificações estiverem completamente corretas, a realização do *software* pode ser muito rápida, mas em projetos de médio e grande porte é muito difícil de acontecer.

Morse (2016) define que o Modelo Cascata é capaz de adaptar as trocas de membros da equipe, o uso do modelo permite que o projeto como um todo possa manter o escopo e a estrutura de *design* mais detalhada e robusta devido a todas as fases iniciais de planejamento e documentação. Sendo comum em projetos com grandes equipes que acabam presenciando membros entrarem e saírem ao longo do desenvolvimento. Assim, permitindo que os deveres do *design* sejam colocados na documentação e decentralizando em qualquer membro da equipe. O modelo em cascata força o projeto e até mesmo a empresa que o utiliza, a ser disciplinado em seu *design* e estrutura. O Modelo Cascata permite a realização de alterações precoces no início do desenvolvimento pelo fato que nas primeiras fases a equipe de desenvolvimento e o cliente estão detalhando os documentos de especificação. E quaisquer alterações podem ser feitas com o mínimo de esforço visto que nenhuma codificação ou implementação foi desenvolvida. E com sua estrutura linear específica a um projeto em cascata são viáveis a organizações ou equipes que trabalham

com marcos e datas. Com estágios bem definidos e assimilados por todos da equipe possam compreender e se organizar, se torna simples em desenvolver cronograma para todo o processo e conceder marcadores e marcos para cada fase e até a conclusão.

2.7.2 As desvantagens do uso do Modelo Cascata (*framework Waterfall*)

Dias (2019) define que o modelo não é flexível, segue uma sequência engessada e é considerado um modelo frágil pelo fato que nos dias de hoje os projetos mudam muito no decorrer do desenvolvimento. Também é difícil estabelecer claramente todos os requisitos logo no início do projeto, pois é natural que no começo do projeto possa existir alguma incerteza. O cliente acaba tendo que aguardar por muito tempo para poder visualizar uma versão executável do *software*, ficando disponível em uma fase avançada do desenvolvimento. Projetos de grande porte raramente seguem a sequência proposta pelo modelo pelo fato que a produção e aprovação de documentos durante as fases se tornam caras e geradoras de muito retrabalho. Durante o processo de desenvolvimento, é normal que o cliente solicite o congelamento de partes do projeto para dar continuidade aos estágios posteriores, Com isso faz com que a solução de problemas fique mais difícil, pois acaba sendo programada para mais tarde e, em algumas das situações acaba sendo ignorada.

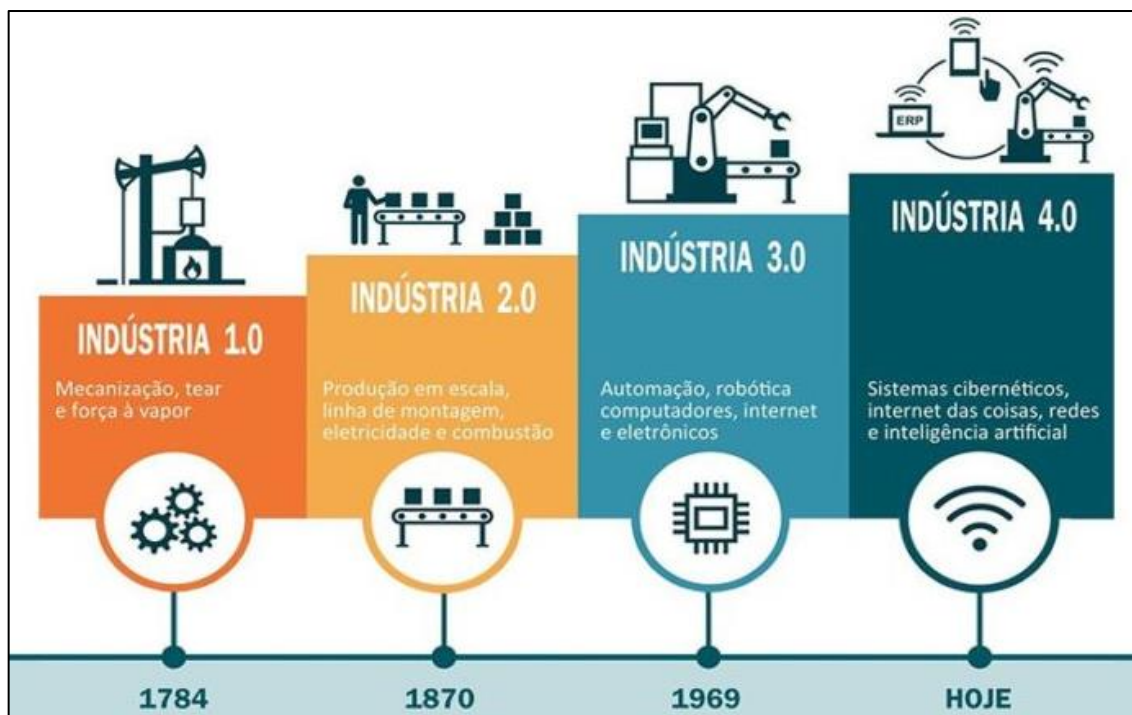
Morse (2016) define que há restrições de *design* não adaptáveis, o aspecto mais prejudicial do modelo em cascata é a falta de adaptação a mudanças em todas as fases de desenvolvimento. Quando ocorre uma sequência de testes na fase de teste e revela uma falha fundamental no *design* do sistema, faz com que tenha que regredir algumas fases do processo, mas em algumas situações pode também fazer com que perca a legitimidade de todo o processo. Outro problema é a dificuldade em contornar é que o *feedback* do usuário ou cliente que é informado no final do ciclo de desenvolvimento e faz com que possa ser tarde ou até mesmo tarde demais. E enquanto os outros modelos mais modernos tentam integrar cada vez mais os testes como um processo fundamental e presente no contexto de desenvolvimento, o Modelo Cascata evita a realização de testes em grande parte até muito tarde.

2.8 Indústria 4.0

Schwab (2016) define que na Alemanha o termo Indústria 4.0 foi divulgado pela primeira vez em 2011 na Feira de Hannover. Com o propósito de descrever como irá revolucionar a organização das cadeias globais de valor. Esse termo refere-se à quarta Revolução Industrial, criando um mundo onde os sistemas físicos e virtuais de fabricação cooperam de forma global e flexível. Entretanto, essa revolução é considerada como uma evolução pelo fato que seu escopo é muito mais amplo. O que a torna diferente das revoluções anteriores é a junção de tecnologias relacionadas ao sequenciamento genético até a nanotecnologia, das energias renováveis a computação quântica, ou seja, há interação entre os domínios físicos, digitais e biológicos. Sendo assim, hoje quando falamos em Indústria 4.0 estamos integrando sistemas de produção inteligentes para trazer mais eficiência e segurança para os mais diferentes setores industriais. E atendendo a vontade dos clientes por produtos e soluções individuais. Desse modo, necessário produzir de forma individual para o cliente e, ainda assim, econômica.

Dreher (2015) define que os termos “*smart factory*”; “*intelligent factory*”; “*factory of the future*” descrevem como serão as indústrias no futuro. A Indústria 4.0 são processos de produção que buscam economizar recursos e energia. Nesta concepção as fábricas serão muito mais inteligentes, flexíveis, dinâmicas e ágeis. Para isso, são necessárias máquinas e instalações modulares e flexíveis para que possam adaptar rapidamente às demandas em constante alteração. Outra definição para “*Smart factory*” é uma fábrica que faz produtos inteligentes, em equipamentos inteligentes, em cadeias de abastecimento inteligentes. Com a produção conectada ao um sistema de TI inteligente máquinas e instalações conseguem se organizar de forma autônoma e independente. Na figura 17 mostra as 4 revoluções Industriais e suas evoluções.

Figura 17 - Representação das 4 revoluções industriais.

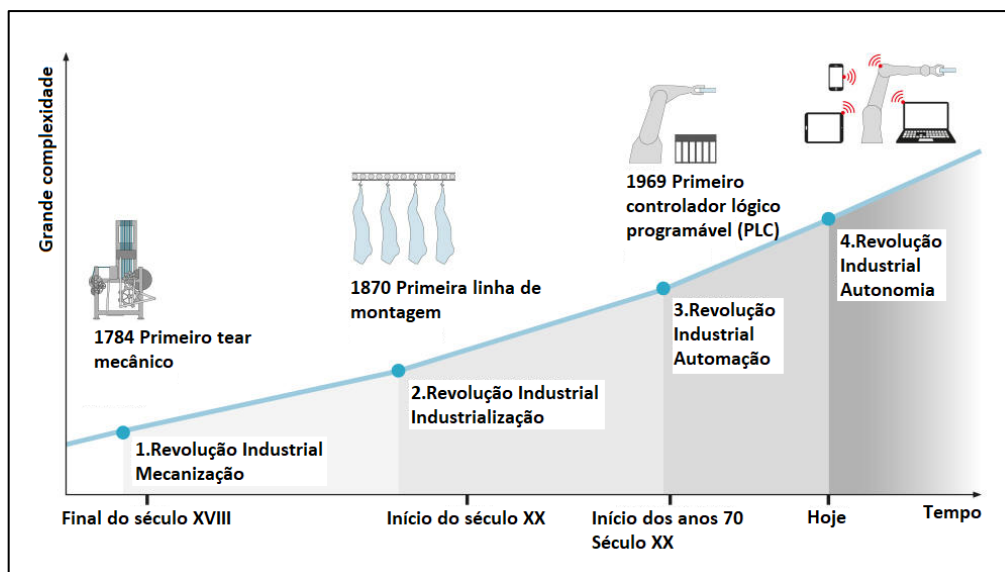


Fonte: Piedade (2019).

2.8.1 Evolução industrial 1.0 a 4.0

O mundo presenciou o nascimento da indústria que conseqüentemente marcou o modo em que a humanidade exercia suas práticas de produção de produtos, dessa forma, a concepção de indústria que conhecemos hoje em dia passou por diversas mudanças durante as motivações e interesses de várias gerações, todas serviram para mudar de forma considerável o desenvolvimento de uma sociedade. Na figura 18 apresenta a evolução da indústria e que ao longo de diversas descobertas e inovações que foram fundamentais para que hoje possamos acompanhar a mais nova transformação tecnológica sem precedentes.

Figura 18 - Evolução da indústria.



Fonte: PILZ (2020).

Bezerra (2020) define que a Primeira Revolução Industrial ocorreu no final do século XVIII e do século XIX. Com a produção artesanal a maioria da população europeia vivia no campo e o produtor dominava todo o processo produtivo. Na estrutura socioeconômica a principal mudança dessa revolução foi à substituição do trabalho artesanal pelo assalariado e com mecanização de máquinas a vapor e o uso da força hidráulica revolucionou a indústria.

A partir do final do século XIX a Segunda Revolução Industrial se estabeleciam as bases do progresso tecnológico e científico, buscando o desenvolvimento e o constante desenvolvimento dos produtos e técnicas. Com a descoberta da energia elétrica, produção em massa, além do emprego do aço, dos combustíveis derivados do petróleo, a invenção do motor a explosão, da locomotiva a vapor e o desenvolvimento de produtos químicos foram as principais inovações para melhor desempenho indústria nesse período.

No século XX a Terceira Revolução Industrial atingiu o ponto culminante do desenvolvimento da eletrônica. Através do surgimento dos componentes eletrônicos e das tecnologias de informação que começaram a espalhar-se na indústria de forma extremamente rápida. O computador, o fax, a engenharia genética, o celular seriam algumas das inovações dessa época.

Venturelli (2017) define que em 2011, o conceito da indústria 4.0 fez com que iniciasse a quarta etapa sendo motivada pelo avanço exponencial da capacidade dos computadores, imensa quantidade de informação digitalizada e novas estratégias de inovação (pessoas, pesquisa e tecnologia). Sendo considerada como uma evolução dos sistemas produtivos devido ao protagonismo da internet foi possível trazer redução de custos, economia de energia, aumento da segurança, conservação ambiental, redução de erros, fim do desperdício, transparência nos negócios, aumento da qualidade de vida e personalização/escala sem precedentes. As tecnologias responsáveis por desenvolverem esse conceito é IoT – *Internet of Things* (Internet das Coisas) e o M2M - *Machine to Machine* (Máquina para Máquina). Ou seja, a indústria 4.0 tem como foco a conectividade, sendo possível conectar toda a indústria, desde a produção até o sistema de vendas.

2.8.2 Pilares da indústria 4.0

No ano de 2012, o conceito de Indústria 4.0 surgiu na Alemanha como um projeto do governo do país buscando assegurar que a tecnologia era um excelente alicerce para a mudança e evolução na concepção de indústria no mundo todo. Com o desenvolvimento em inovações tecnológicas nos campos de automação e tecnologia da informação, teve como objetivo de desenvolver processos mais rápidos, flexíveis e eficientes. A indústria 4.0 permite coletar e analisar dados entre máquinas, podendo gerar processos mais rápidos e eficientes e gerando produtos com qualidades melhores com custos menores. Melo (2019) define que a indústria 4.0 é baseada em princípios e pilares. Com o objetivo de conectar máquinas, sistemas e ativos, as empresas poderão criar redes inteligentes em todas as áreas da empresa para que possa controlar os módulos da produção de forma autônoma. Na figura 19 apresenta os nove pilares da indústria 4.0 sendo análise de dados, robótica, simulação, integração de sistemas, internet das coisas (IoT), cibersegurança, *cloud computing*, manufatura aditiva e realidade aumentada.

Figura 19 - Representação dos nove pilares da indústria 4.0.



Fonte: Altus Evolução em Automação (2019).

Almeida (2019) define que a Conferência das Nações Unidas sobre Comércio e Desenvolvimento (UNCTAD) destaca que adquirir cada vez mais dados permite diversos benefícios econômicos devido ao fato de que sua utilização facilita sua transformação em um ativo rentável. Conseqüentemente, com o uso desses dados faz com que se tenha um aumento da produtividade e da lucratividade. Também fornece financiamento adicional para adquirir bancos de dados ou *softwares* complementares. Para as empresas que se dispuseram a investir nesse segmento dará vantagem de ampliar o valor de seus dados e a base de conhecimento associada.

Garcia (2012) define que os robôs autônomos são denotados como máquinas inteligentes e que são preparados para realizar várias atividades com nível alto de complexidade que proporcionam resultados que os humanos não são capazes de produzir. Os robôs conseguem reduzir o tempo de produção com ciclos cada vez mais rápidos e repetitivos, capaz de coletar diversas informações em seu entorno e podendo trabalhar de forma independente. Assim, sendo capaz de adaptar a novas condições ou ajustar estratégias para a realização de tarefas como a precisão na manipulação de micropeças.

Abreu (2017) define que simulação é uma técnica que utiliza a modelagem baseada em um sistema computacional para imitar certos aspectos da realidade, o que permite trabalhar em condições semelhantes às reais. Sendo possível analisar

previamente todas as etapas, mas com variáveis controladas em um ambiente próximo da realidade. Assim, facilitando a visualização de erros, custo-benefício e tempo. Esta técnica permite que os processos e produtos sejam testados e ensaiados durante a fase de concepção, reduzindo custos com falhas e o tempo de projeto (Vitali, 2018, p.1).

ARKTIS (2016) define que a integração de sistemas é uma técnica que promove a unificação dos sistemas de gerenciamento e controle buscando melhoria no desempenho e na organização das operações de modo a conectar diretamente o chão de fábrica com nível corporativo. Além disso, essa integração verticalizada é capaz de aumentar a competitividade, redução de custos e o aumento da flexibilização da produção. Assim, a integração de sistemas permite que sejam empregadas em vários setores e não ficará limitada por um único espaço físico, interligando cadeias logísticas, fornecedores, fabricantes e consumidores, gerando possibilidades de abertura de novos negócios, melhoria e desenvolvimento de novos produtos.

Coelho (2016) define que o termo IoT – *Internet of Things* (Internet das Coisas) surgiu no ano de 1999 pelo pesquisador britânico Kevin Ashton que desenvolvia sua pesquisa na área de identificação por rádio frequência (RFID). Desde então, tem sido impulsionada pelo potencial de agregar valor a processos produtivos, logística, entretenimento, comunicação, na eficiência dos serviços da área da saúde, educação, energia e entre muitas outras áreas.

Giraldo (2018) define que o termo *Cloud Computing* conhecida também como computação em nuvem, é a tecnologia que possibilita de forma remota o uso de recursos da computação por meio da conectividade da internet. As informações então numa nuvem compartilhada que permite que o usuário acesso por meio de qualquer aparelho eletrônico capaz de acessar a internet. A opção por serviço na nuvem permite a redução de custos com infraestrutura, economia do espaço, centralização da informação, trabalho remoto e o aumento ou diminuição de gastos acordo com a necessidade do cliente.

Antoniuzzi (2017) define que o termo cibersegurança ou segurança cibernética é responsável por lidar com os riscos oriundos do cyber espaço que está intrinsecamente vinculado com a internet. As comunicações entre máquinas e dispositivos estão associadas a códigos maliciosos incorporados nos dispositivos portáteis, o código malicioso incorporado dentro de um aparelho eletrônico em um ambiente to-

do conectado seria considerado um risco de cyber espaço. Segurança cibernética inclui os controles defensivos que são necessários para lidar com a ameaça do cyber espaço. Junto com a evolução das tecnologias os *cybers* crimes evoluíram e criaram uma ameaça na gestão de riscos, tornando defasadas as ferramentas e metodologias tradicionais de segurança da informação, forçando as empresas a revisarem todos os aspectos organizacionais, procedimentos, políticos e prioridades.

Stefani (2014) define que as impressoras 3D foram criadas a mais de 30 anos, mas recentemente o seu uso se tornou popular por causa do avanço tecnológico que fez com que os insumos ficassem mais acessíveis. Manufatura aditiva ou impressão 3D é a tecnologia conhecida como a evolução da prototipagem, atualmente diversas empresas e profissionais utilizam as impressoras para poderem criar protótipos de peças e até peças finais para diversos fins, assim, sendo possível a criar qualquer objeto a partir de um arquivo digital e ser montado por uma impressora 3D. Isso possibilita uma grande redução nos custos de fabricação, como por exemplo, não precisando gastar na fabricação de moldes, usinagem ou outro processo custoso.

Paiola (2019) define que a realidade aumentada pode ser aplicada na segurança do trabalho, treinamento e capacitação, manutenção industrial e automação de processos na linha de produção. Na segurança do trabalho a realidade aumentada possibilita para os responsáveis verificarem se os operadores estão utilizando os equipamentos de segurança (EPI) de forma correta e em quais condições estão realizando as operações de trabalho. Para treinamentos a utilização de óculos de realidade aumentada, *tablets* ou *smartphones* se torna possível ampliar a capacidade de aprendizado e retenção por parte dos funcionários, uma vez que essa tecnologia promove a interatividade como principal vantagem. Na manutenção industrial a utilização desse tipo de tecnologia abre a possibilidade de serem auxiliados remotamente por especialistas e a visualização de protótipos e instruções diretamente no display do dispositivo. Por fim, na automação de processos industriais está intrinsecamente ligada à capacidade de resolução de problemas de forma ágil e eficaz.

2.8.3 Indústria 4.0 no mundo

Coelho (2016) define que o impacto causado pela Indústria 4.0 vai além da simples digitalização das empresas, estamos passando por diversas mudanças de

forma muito mais complexa relacionada à inovação baseada em múltiplas tecnologias combinadas em todos os setores produtivos, que forçarão as empresas a repensarem na forma de como administrar seus projetos, negócios, processos, posicionamento na cadeia de valor (competitividade), na forma de desenvolver novos produtos e introduzi-los no mercado, ajustando as ações de marketing e de distribuição.

Venâncio (2017) define que a Indústria 4.0 tem a capacidade de evoluir de forma exponencial o parque industrial desde o aspecto produtivo até a conexão da cadeia logística. A possibilidade de transformar sistemas industriais em *cyber-physical systems* (CPS), será possível transformar toda relação de trabalho e de serviços de uma indústria de maneira definitiva. Sistemas totalmente integrados em tempo real sendo conectados entre *software* e *hardware*, com sensores enviando sinais para cópias virtuais das máquinas físicas e podendo armazenar uma grande quantidade de informações. E ainda também será possível prever falhas e potencializar processos garantindo que o futuro da manufatura seja capaz de derrubar uma tecnologia já preestabelecida no mercado, representando a tendência industrial transformando toda a cadeia de trabalho e de serviços de uma indústria de maneira definitiva.

A posição de liderança internacional da Alemanha em sistemas integrados, em soluções de segurança e em *software* empresarial, aliada a uma invejável reputação de engenharia em questões relacionadas a soluções de sistemas, tecnologias semânticas e *know-how* de sistemas integrados, tornou possível que o país assumisse um papel pioneiro no desenvolvimento de sistemas ciber-físicos (CPS), que fornecem a base para a criação da *Internet* das Coisas, a qual combinada com a *Internet* dos Dados e dos Serviços torna possível a Indústria 4.0. (IEDI, 2017, p.1).

Campos (2018) define que entre todos os efeitos possíveis estão ainda a modificação dos métodos de trabalho, com a utilização de sistemas e sensores inteligentes fazem com que os *softwares* rígidos e centralizados percam espaço para recursos de inteligência artificial e comunicação entre máquinas (M2M) no chão de fábrica. Além disso, abre a possibilidade para que as empresas possam realizar produções de mercadorias de acordo com demanda ou a necessidades de cada cliente, garantindo maior satisfação e fidelização por meio do método de customização.

Gonçalves (2016) define que estamos presenciando uma era em que a comunicação se torna cada vez mais globalizada e conectada em nossa sociedade. As dificuldades que existiam para que pudéssemos nos comunicar com pessoas de

diferentes lugares do mundo se tornou cada vez mais acessível e estreitando as distancias, assim, diminuindo as barreiras que dificultam a globalização e aumentando o poder de se conectar em qualquer lugar e a qualquer hora. Isso faz com que se torne de grande importância e utilidade para as indústrias, pelo fato que as empresas do futuro utilizarão não só para a comunicação entre pessoas, mas também estenderá as máquinas, sensores e atuadores, ou seja, nesse contexto, a chamada “indústria inteligente”, com todas as suas máquinas e insumos, dialogam e trocam dados entre si ao longo das operações industriais. E sendo importante para garantir o funcionamento de operações em diversos segmentos e a otimização de custos em toda a cadeia produtiva. Com isso, as empresas que conseguirem se adaptar melhor e o mais rápido possível a este novo modelo de polo industrial estará credenciada a continuarem competitivas e ampliando suas chances dentro do mercado consumidor.

Silva (2017) relata que nos EUA, a Indústria 4.0 está de fato tendo um impacto muito importante nas iniciativas de fábricas inteligentes (*smart factories*), especialmente com a colaboração com o Consórcio Industrial da *Internet*. A realidade 4.0 não se trata apenas de fábricas com alto grau de automação.

2.8.4 Indústria 4.0 no Brasil

Rodrigues (2018) relata que as indústrias brasileiras ainda estão entre a indústria 2.0 e indústria 3.0. E têm como desafio evoluir rapidamente para indústria 4.0, sob a ameaça de perder completamente a competitividade no mercado globalizado.

IEDI (2018) Instituto de Estudos para o Desenvolvimento Industrial relata que se o Brasil não priorizar este tema e acelerar a formulação de um plano nacional, inclusive articulando os diferentes agentes de seu sistema de inovação, sua posição no ranking global da indústria, que já está em rota descendente, pode recuar ainda mais. Em 2016, o país encontrava-se na 9ª colocação, isto é, muito próximo de ser excluídos do grupo das dez maiores potências industriais.

Sena (2017) relata que a Indústria 4.0 desponta como caminho natural para aumentar a competitividade do setor por meio das tecnologias digitais. No Brasil ainda é pouco utilizada pelas empresas nacionais. O atraso brasileiro diante da integração das tecnologias físicas e digitais em todas as etapas de desenvolvimento de um

produto fica evidente porque a maioria das empresas não identificam quais tecnologias têm potencial para alavancar a competitividade do setor industrial.

CNI (2018) relata que poucas empresas estarão preparadas para enfrentar todas estas mudanças de uma vez. Existem, por outro lado, milhares de empresas que deverão participar do processo de difusão dessas novas tecnologias paulatinamente, de acordo com suas trajetórias, suas capacitações e suas estratégias. Nesse contexto, o foco de uma iniciativa visando ao desenvolvimento da Indústria 4.0 no Brasil deve ser o de empresas que mais cedo entrarão no novo paradigma e estimular as demais a apressarem sua inserção na nova onda, sob risco de não conseguirem sobreviver no novo ambiente competitivo.

Segundo a revista Exame (2016) afirma que muitas indústrias brasileiras já automatizaram seus processos, mas ainda não alcançaram a manufatura digital.

A indústria 4.0 é composta por duas vertentes: processos integrados que garantem a produção customizada e produtos inovadores. O Brasil precisa ainda andar muito nesses dois sentidos. Temos poucos setores competitivos em escala global, afirma o professor Eduardo de Senzi Zancul, da Escola Politécnica da Universidade de São Paulo. (Revista Exame, 2016, p.1).

Alcântara (2017) relata que algumas indústrias brasileiras saíram na frente, com projetos que podem ser considerados 4.0. Veja o caso da Ambev. Em 2015, a multinacional de bebidas adotou um sistema de automação para melhorar o controle do processo de resfriamento da cerveja e reduzir as variações de temperatura, evitando, assim, o desperdício de energia. A tecnologia já está em oito cervejarias da empresa e a previsão é expandir o uso para outras unidades ao longo deste ano.

Na Volkswagen Brasil, todos os projetos nascem a partir de um modelo digital. Os produtos são simulados em ambiente 3D, o que acelera o processo, garante flexibilidade, otimiza o tempo de produção e ainda abre postos de trabalho altamente qualificados. A Volkswagen tem investido em software, hardware e treinamento para que os funcionários passem a lidar com essa nova realidade. Cinco novas iniciativas nas fábricas brasileiras já permitiram uma economia para a empresa de 93 milhões de reais em dois anos. (Revista Exame, 2016, p.1).

Ribeiro (2019) relata que o Brasil se mostra empenhado em investir nas empresas, criando programas de incentivo para a evolução das indústrias Agência Brasileiros de Desenvolvimento Industrial (ABDI), mas precisa de uma maior divulgação dos programas e possibilidades para que as empresas tomem conhecimento e possam se planejar melhor. Os profissionais por sua vez, precisam melhorar suas habilidades e capacitar-se com as habilidades ditas como importantes no cenário atual caso queiram permanecer no mercado de trabalho.

3 METODOLOGIA

Neste capítulo são explorados estudos sobre o uso de ferramentas ágeis, mais especificamente no uso do *framework* Scrum no desenvolvimento de *software* automotivo. Ressaltam-se a relação da nova Revolução Industrial que digitalizou as indústrias do Brasil e do mundo, tendo um impacto muito importante na introdução de fábricas inteligentes. Essa revolução é considerada uma evolução para todas as áreas produtivas e para acompanhar essa evolução é necessário refletir na forma de gerenciar os projetos para que possam não conflitar com a necessidade de atender a vontade dos clientes por produtos e soluções individuais, e ainda sendo competitivo. Iremos apresentar as vantagens e desvantagens do método ainda mais utilizado nos dias de hoje entre as montadoras, o Modelo em V. Embora esse modelo traga uma experiência positiva infelizmente ela não é perfeita. Com isso, será apresentado como introduzir uma ferramenta ágil como o Scrum para suprir todas as necessidades e ainda atender onde o Modelo em V não consegue desempenhar grandes resultados devido a sua concepção.

Algumas das informações apresentadas como suporte às análises provêm de estudos publicados entre empresas do setor automotivo, mas também de fontes acadêmicas que nos auxiliam no desenvolvimento e no entendimento da viabilidade da utilização do Scrum. Tais informações oriundas das empresas são apresentadas de forma informativa. Decorrente do embasamento na fundamentação teórica deu-se início o desenvolvimento do projeto seguindo algumas etapas pré-estabelecidas pelo orientador e o orientado. A proposta deste trabalho é apresentar o *framework* Scrum para quem ainda não o conhecem e mostrar que as ferramentas ágeis estão se tornando cada vez mais importantes para o gerenciamento e desenvolvimento de projetos automotivos, mas também está sendo utilizado em outros segmentos produtivos. O projeto apresentará a utilidade e viabilidade no segmento automotivo com exemplos, e também em outros segmentos e *cases* de sucessos.

3.1 Qual relação entre *framework* Scrum e Indústria 4.0

Indústria 4.0 está realizando uma transformação significativa nas plantas fabris gerando um grande impacto no conceito de produtividade, causando reduções de custos, controle sobre o processo produtivo, controle do consumo de matéria pri-

ma, customização da produção e a competitividade no mercado consumidor. Strafacci (2019) define que a metodologia ágil vai de encontro com a flexibilidade, rapidez e segurança exigidas pela Indústria 4.0, oferecendo a capacidade de realizar ajustes no projeto e no produto conforme eles evoluem. Ao considerar as mudanças rápidas do ambiente como os clientes, a tecnologia e os concorrentes da gestão de projetos vai se adaptando de forma flexível, gerando aprendizado durante o processo. Isso garante o melhor resultado possível, de forma eficiente e sustentável para a empresa, ao mesmo tempo em que maximiza a experiência do cliente em cada etapa.

Essas mudanças causadas na forma de produzir faz com que afete primeiramente e diretamente na forma de pensar em como gerenciar o desenvolvimento dos projetos, com prazos para implementar cada vez mais reduzidos e com uma tolerância muito menor para erros, começou a questionar os métodos de gestão tradicional de projetos que são empregados até hoje. Os métodos convencionais começaram a apresentar falhas em seu desenvolvimento a partir do momento em que precisam de longos processos de desenvolvimento e criação para poder alcançar seus objetivos e lançar seus produtos ao consumidor final, assim, diminuindo a capacidade das empresas de se tornarem competitivas.

Com essa deficiência, os métodos ágeis estão se tornando cada vez mais importantes nesse período de mudança na forma de produzir e pensar, eles podem ser usados para evitar longos processos de desenvolvimento e auxiliar os fabricantes a lançarem seus produtos mais rapidamente. Cada vez mais, as indústrias estão adotando métodos ágeis para facilitar os processos dentro da linha de produção. Vinal (2018) define que o desenvolvimento dos métodos inteligentes foi a solução para desenvolver uma execução correta das ações, de forma que o objetivo final da empresa seja atendido no prazo estipulado. Com o objeto de simplificar a maneira em que os projetos são executados, impactando positivamente no resultado final.

Strafacci (2018) define que esses dois movimentos são sinérgicos, as mudanças constantes pela Indústria 4.0 vão além de *software* e *hardware*, passando pela construção da estratégia, processos, metodologias, indicadores e a mudança de mentalidade que diz respeito principalmente à gestão de projetos que podem (e devem) seguir um ciclo ágil. As entregas em eventos menos, em última instância, reduzem custos, pelo fato que há diminuição de falhas e geram encurtamento da curva de aprendizado. Também tornam a empresa mais versátil nas suas escolhas estra-

tégicas, ou seja, no final das contas, permite que a indústria identifique gargalos e consiga eliminá-los.

Vinal (2018) define que o *framework* Scrum permite que as empresas possam criar soluções inovadoras para o mercado consumidor, fazendo com que se tenha um aumento da competitividade. Essa ferramenta de gerenciamento ágil existe devido à necessidade de oferecer agilidade em projetos de alta complexidade e urgência de entrega com o maior valor possível. O Scrum, por trabalhar com um conjunto de ideias não expressas de forma explícita de como solucionar o problema, permite que o desenvolvedor possa focar seus esforços na resolução do problema.

Santos (2018) define que Indústrias são consideradas com altíssima eficiência quando todos os processos acontecem sem falhas ou desperdícios de material, com foco na redução constante de custos na produção, para que os responsáveis desse setor possam constantemente aprimorar o serviço entregue ao consumidor final. É essencial que esses gestores acompanhem o passo a passo da fabricação do seu produto.

No segmento automotivo a Indústria 4.0 e o uso do *framework* Scrum pode contribuir significativamente no aumento da qualidade e na redução de custos, com o uso de equipes de desenvolvimento ágeis novas soluções e produtos podem ser implementados dentro do prazo e podendo reconfigurar seu processo de coleta de dados de forma que possam ser revertidos em *feedback*. Strafacci (2018) define que com a aprovação em 2018 da Rota 2030, o novo programa poderá incentivar, não só a produção de veículos mais eficientes e seguros, mas a implantação da Indústria 4.0. Essa pode ser considerada uma ação fundamental para tornar as indústrias do setor automotivo competitivas.

3.1.1 As vantagens do uso do *framework* Scrum

Lopes (2017) define que a maior vantagem de introduzir o uso de ferramentas ágeis como o *framework* Scrum no gerenciamento de projetos é a simplicidade. No desenvolvimento de um projeto com o gerenciamento ágil é possível definir claramente os papéis e, além disso, partes do produto final podem ser entregues ou apresentadas em ciclos rápidos de iteração, sendo bem utilizado em cenários onde é necessário rapidez e flexibilidade. Os métodos tradicionais de gerenciamento de pro-

jetos são melhores utilizados quando o escopo do projeto está definido por completo no início do projeto e não é necessário gerar alterações ao longo do desenvolvimento. O Scrum possui uma estrutura que está mais adequada à realidade inconstante trazendo mais benefícios como, por exemplo, a percepção da diminuição do tempo de consumo nos projetos.

Contezini (2017) define que o *framework* Scrum possui quatro vantagens no uso para gerir projetos. As vantagens são:

- **Realização:** A capacidade de concluir as atividades estabelecidas em uma *Sprint* faz com que a equipe fique com a sensação de realização e de conclusão dos trabalhos a cada entrega e isso auxiliar em manter a motivação da equipe ao longo do desenvolvimento do projeto.
- **Transparência:** Dentro da organização do Scrum prevê que o projeto pode ser observado por todos que fazem parte ou que fazem parte da organização. Essa vantagem não é observada em outras propostas de gestão de projetos.
- **Ausência significativa de falhas:** Com o foco em realizar entregas com qualidade faz com que diminua drasticamente a quantidade de falhas e erros. Assim, a qualidade é mais importante do que o prazo de entrega.
- **Reordenação:** O Scrum permite que a equipe de desenvolvimento possa fracionar o projeto e manejar as prioridades de acordo com o andamento de cada *Sprint*, e sendo possível concentrar os esforços necessários para finalizar etapas que ainda não foram concluídas.

Duarte (2017) define que o Scrum possui nove vantagens ligadas diretamente ao uso quando aplicado no time de desenvolvimento, sendo adaptabilidade, transparência, *feedback* contínuo, melhoria contínua, entrega contínua de valor, eficiência, motivação, alta velocidade e ambiente inovador.

- A adaptabilidade faz com que o Scrum consiga lidar com a imprevisibilidade que podem surgir ao longo de projetos longos ao inserir mais resiliência na gestão do projeto.
- A transparência é um dos pilares do *framework* Scrum e na ausência de transparência não há inspeção adequada, adaptação, engajamento, confiança, evolução do time e sucesso no projeto.

- O *feedback* contínuo fornece informações importantes referente a performance de cada pessoa do time, orientando se as ações e os trabalhos realizados estão gerando o resultado esperado, auxiliando em deixar a equipe alinhada em saber como podem ajudar uns aos outros. O *feedback* contínuo é fornecido através dos eventos como a *Daily Scrum* e *Sprint Review*, assim, garantindo o segundo pilar do Scrum, a inspeção que leva a adaptação, em um ciclo virtuoso que gera melhoria contínua.
- O Scrum ensina que a inspeção vai te levar inerentemente ao aprendizado e com isso a melhoria contínua dos processos e produtos. As entregas realizam melhorias contínuas de forma progressiva no incremento de cada *Sprint* através do processo de aperfeiçoamento do *Product Backlog* e do processo em si.
- A realização de entregas contínuas em cada iteração do Scrum faz com que um incremento do produto seja entregue ao cliente e gerando valor a cada entrega. O processo de priorizar e criar o *Product Backlog* garante o cumprimento das exigências de mais valor no início do desenvolvimento do projeto para que sejam apresentadas aos clientes. Essa abordagem colaborativa com o *stakeholder* e o foco no valor do negócio garante uma estrutura focada na satisfação do cliente.
- A eficiência do Scrum é representada pelos *Sprints*, cada um deles é representado por um *time boxing* que se encontra um conjunto de atividades a serem realizadas. O *time boxing* é a minimização de trabalho não essencial que faz com que conduza a níveis mais altos de eficiência.
- A motivação conduz os times de desenvolvimento se tornar cada vez mais confiantes devido à transparência e resultando em um ambiente de trabalho de alta confiança. Uma vez que a equipe tem o suporte e apoio necessários para que possa avançar e desenvolver sem impedimentos é possível atingir *a níveis mais altos de motivação entre os colaboradores*.
- A alta velocidade é uma vantagem da estrutura do Scrum que permite que as equipes multifuncionais possam atingir seu pleno potencial de elaboração de entregas em menor tempo com maior valor de forma consistente. Por exemplo, a alta velocidade obtida pelo Scrum não está na quantidade de linhas escritas em cada *Sprint*, mas nas linhas certas que serão escritas e consequen-

temente serão as que trarão maior resultado para o cliente em menor tempo, assim, assegura que o cliente sempre receba um alto retorno do investimento em *software*.

- O Scrum proporciona um ambiente inovador devido à realização da *Sprint Retrospective* e da *Sprint Review* que criam um ambiente de aprendizagem e adaptabilidade para melhorar a forma como as coisas devem ser feitas, levando o desenvolvimento do projeto para o caminho correto e não conforme foi planejado anteriormente, assim, tendo a chance de visitar o processo e ajustá-lo a cada *Sprint*.

3.1.2 As desvantagens do uso do *framework* Scrum

Lamelas (2018) define que as desvantagens do framework Scrum são a visão segmentada do projeto e a procura de ser ágil faz com que os desenvolvedores possam perder a perspectiva do projeto como um todo, com o foco em apenas uma parte do desenvolvimento podem causar falhas no momento em que é realizada a junção das partes para concluir o projeto. E a desordem nas funções pode causar confusão em alguns membros da equipe de desenvolvimento devido à presença de papéis indefinidos nas funções presentes no projeto.

Schermann (2018) define que a maior desvantagem do Scrum é a dificuldade em seguir a estrutura que o framework oferece e conseqüentemente muitos projetos acabam fracassando devido à falta de conhecimento técnico, resistência por parte do time de desenvolvimento, falta de costume de trabalhar com ferramentas ágeis e por falta de referências de projetos.

3.2 Como é o gerenciamento convencional na indústria automotiva atualmente

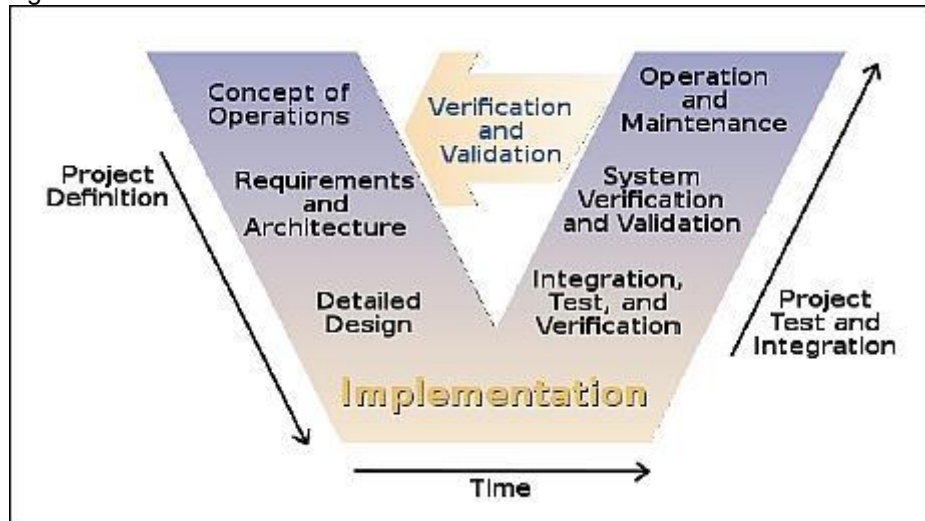
Para o desenvolvimento de um produto automotivo uma das principais abordagens aplicadas ao desenvolvimento baseado em *software* é o Modelo em V em função dos processos e relações entre componentes, à integração do sistema e à realização do projeto. O Modelo em V é a abordagem padrão porque é simples, fácil de usar e suporta *design*, planejamento, desenvolvimento, integração e verificação. Este modelo permite que os fabricantes desenvolvam simultaneamente peças de veículos através de diferentes fornecedores por meio de normas e especificações

congeladas no início do desenvolvimento. Santos (2018) define que hoje em dia, é viável virtualizar quase todo o processo de desenvolvimento e produção do veículo devido a grande quantidade de ferramentas de virtualização existentes. O uso da virtualização é para que não tenha elevadas quantidades de protótipos e sendo possível executar testes automatizados de forma eficaz.

Opencadd (2018) define que para sistemas automotivos, o processo de desenvolvimento é visualizado pelo Modelo em V, e faz com que leve à escolha correta dos componentes, à integração do sistema e à realização do projeto. Para a realização da fase do Modelo em V é preciso lidar com a engenharia de requisitos para o desenvolvimento, isto é, os requisitos fundamentais para um projeto serão coletados em relação às necessidades e limitações do sistema das partes interessadas e após a análise detalhada dos requisitos do sistema.

Morse (2016) define que o Modelo em V representa os diversos estágios que serão passados durante todo o processo de desenvolvimento de *software*. Na figura 20 representa o início que está localizado no estágio superior esquerdo e trabalhando ao longo do avanço em direção à ponta superior direita. Os estágios representam uma evolução linear semelhante ao Modelo Cascata.

Figura 20 - Modelo de Ciclo-V de desenvolvimento.

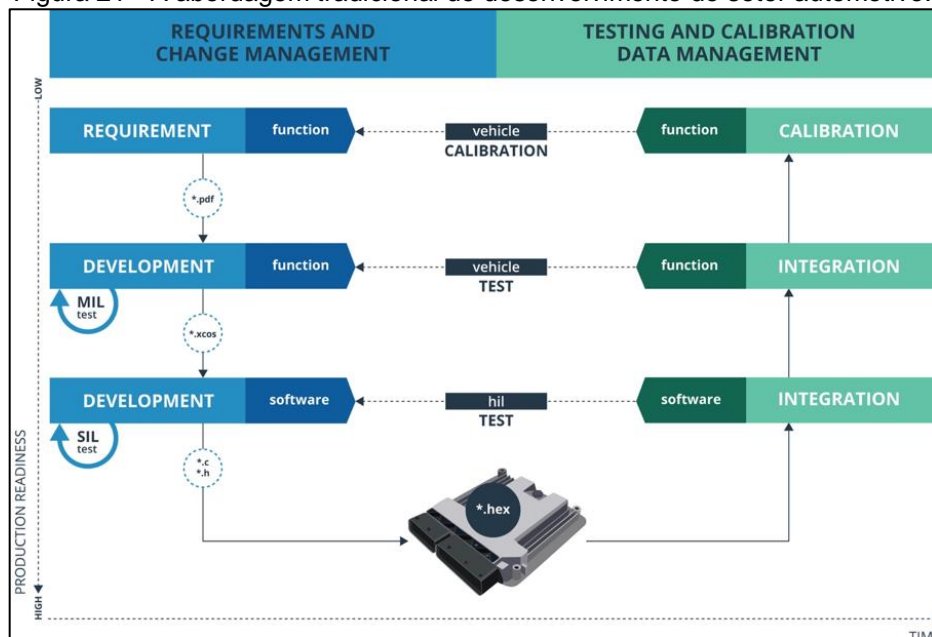


Fonte: Portal Directory (2002).

Firesmith (2013) define que o Modelo em V é baseado no Modelo Cascata enfatizando a verificação e validação. O Modelo em V utiliza a metade inferior do modelo em cascata e o dobra para cima realizando o formato em V, para que seja pos-

sível realizar a verificação ou validação dos produtos de trabalho da direita com os da esquerda. O lado esquerdo de um Modelo em V se concentra na análise de requisitos, arquitetura e *design* e desenvolvimento de *software*, enquanto o lado direito se concentra na verificação e validação do produto em todas as etapas do processo de desenvolvimento. Ou seja, o lado esquerdo do V compõe as atividades de análise que decompõem os requisitos e as necessidades dos usuários em partes pequenas e gerenciáveis, o desenvolvedor poderá executar diversos testes que são chamados de *Model in the Loop* (MiL) que utiliza diagramas em bloco para poder simular as funcionalidades do projeto, enquanto o lado direito do V representa as atividades de síntese correspondentes que agregam ao desenvolvimento do *software* utilizando as técnicas de testes *Software in the Loop* (SiL). Na figura 21 apresenta a abordagem tradicional de desenvolvimento do setor automotivo.

Figura 21 - A abordagem tradicional de desenvolvimento do setor automotivo.



Fonte: Vasytkov (2019).

Silva (2019) define que há áreas/departamentos na indústria automotiva que “acompanham” o Modelo em V, com equipes dedicadas a atender cada etapa. Abaixo segue alguns exemplos de relacionados a áreas de desenvolvimento de *software* automotivo:

- *Systems Requirements*: requisitos de alto-nível, percebidos pelo cliente;

- *Controls Requirements*: requisitos de baixo-nível, que traduzem os requisitos de *Systems* em algoritmos, para então implementação em *software*;
- *Software Development*: responsável por implementar o *software* propriamente dito, a partir dos requisitos do time de *Controls Requirements*;
- *Controls Verification*: de posse do *software*, é responsável por verificar se os requisitos de *Controls* foram atendidos, por meio de testes funcionais; e
- *Systems Validation*: de posse do *software* verificado e calibrado, responsável por validar os requisitos de *Systems*, por meio de testes integrativos.

3.2.1 As vantagens do uso do Modelo em V

Morse (2016) define que devido à concepção rigorosa do Modelo em V e suas fases de projeto, implementação e testes lineares é ideal para ser utilizada em projetos restritos em que o escopo é bem definido, a tecnologia é estável e as especificações de documentação são claras. O uso deste modelo também é viável para o gerenciamento de tempo, sendo utilizado em projetos em que é necessário manter prazos rigorosos e atendendo datas durante todo o processo. Com estágios claros e bem definidos a equipe de desenvolvimento pode entender e se preparar com mais facilidade, tornando relativamente fácil em criar uma linha do tempo para todo o período de desenvolvimento enquanto gera marcos para cada estágio durante o desenvolvimento.

Firesmith (2013) define que o lado positivo do Modelo em V é que consegue representar de forma clara as atividades primárias de engenharia em um fluxo lógico que é fácil de entender e equilibra as atividades de desenvolvimento com as atividades de teste.

3.2.2 As desvantagens do uso do Modelo em V

Morse (2016) define que o Modelo em V tem falta de adaptabilidade, é um problema semelhante encontrado no Modelo Cascata no qual o Modelo em V é baseado. O maior problema deste modelo é a incapacidade de se adaptar a alterações durante o período de desenvolvimento. Os projetos não podem ser facilmente alterados. Portanto, é pouco adequado para ser utilizado em projetos de longo prazo

que podem exigir muitas versões, atualizações e revisões. Outro problema é que a realização de testes no final do ciclo de vida, ou seja, é possível que aconteça a realização de testes de maneira apressada para atender um determinado prazo ou marco. E incentiva o desenvolvimento “elaborado por comitê” visto que a natureza rigorosa e metódica do Modelo em V e seus vários estágios lineares tendem a reforçar um ciclo de desenvolvimento apropriado para gerentes e usuários, em vez de desenvolvedores e *designers*.

Firmesmith (2013) define que o lado negativo do Modelo em V é ser uma simplificação grosseira e excessiva das atividades primárias de engenharia ilustradas como fases sequenciais, em vez de atividades que normalmente acontecem de forma incremental, iterativa e simultânea. Como ocorre em projetos que utilizam abordagens de desenvolvimento evolutivo e ágil. Outro problema é que a distinção entre unidade, integração e teste do sistema não é tão clara quanto o modelo implica.

3.3 Como introduzir o Scrum no desenvolvimento ágil na indústria automotiva

De fato o Modelo em V é o padrão utilizado para o desenvolvimento de *software* automotivo, dividindo o *software* em duas fases principais ou verticais, sendo a primeira fase o gerenciamento de requisitos e mudanças, e a segunda fase a gestão de dados. Embora esse modelo ofereça uma experiência positiva para o usuário, ela não é perfeita. Uma de suas principais desvantagens é a falta de eficiência, fazendo com que leve muito tempo para ir do estágio de pesquisa de requisitos até o *feedback* real do uso do produto. Segundo Vasytkov (2019, p. 1) “Metodologias ágeis podem ajudar a reduzir esse período de espera e melhorar a eficiência operacional do desenvolvimento automotivo”.

Entre tanto, as ferramentas ágeis podem reduzir esse período de espera e melhorar a eficiência do desenvolvimento automotivo. Atualmente, ser ágil significa ser flexível e responder às mudanças diárias de forma eficaz. Com o desenvolvimento ágil de *software* torna-se uma abordagem ágil que auxilia as empresas a obter sucesso em um ambiente em constante mudança. O desenvolvimento ágil consiste em iterações curtas de desenvolvimento, *feedback* de cliente durante todo o fluxo de trabalho para melhorar a eficiência, e compor de equipes multifuncionais e auto-organizáveis.

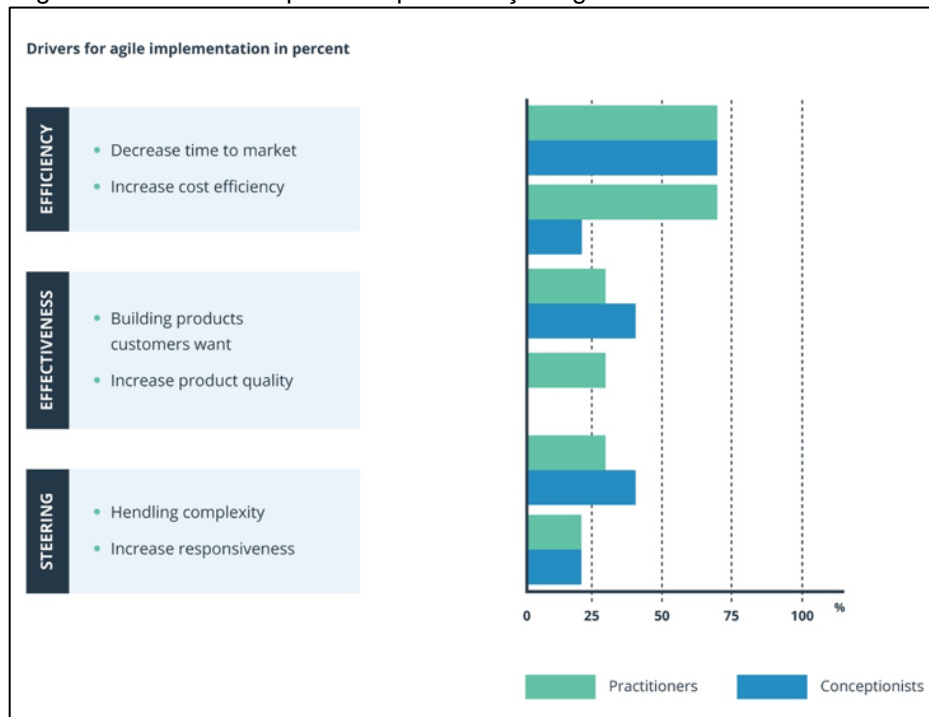
O Scrum fornece uma estrutura para reagir com flexibilidade em mercados voláteis. O objetivo é focar nos benefícios do cliente, evitando gastos desnecessários e investimentos de tempo. As equipes de desenvolvimento se organizam em uma estrutura para coordenar as atividades e trabalhar em conjunto para uma tomada de decisão rápida. (KUGLER MAAG CIE, 2019, p.1).

Os métodos ágeis estão se tornando cada vez mais importantes diante dos crescentes requisitos da indústria automotiva. Podendo ser utilizados para evitar longos processos de desenvolvimento e ajudar os fabricantes de automóveis a lançar seus produtos mais rapidamente. Isso aumenta a capacidade das empresas tornarem mais competitivas.

A agilidade faz parte da história de sucesso na indústria automotiva: até recentemente, os processos, métodos e práticas de desenvolvimento ágil eram frequentemente rejeitados para o desenvolvimento de veículos, agora o setor atingiu certa maturidade na seleção e no uso de abordagens ágeis. (KUGLER MAAG CIE, 2015, p.1).

Vasytkov (2019) relata que o setor automotivo está em meio há uma transformação e precisam mudar a forma de desenvolver e fabricar seus veículos. Nesse período há necessidade de confiar na utilidade de *software* tanto quanto na mecânica. Os veículos de hoje em dia rodam mais em código do que em potência, e as montadoras não podem ignorar esse período transitório. Agilidade é uma característica essencial que as montadoras devem adotar na era digital e conectada. Na figura 22 segue os três grupos de fatores para a implementação ágil, sendo eficiência, eficácia e direção. Em porcentagem, apresentando os valores entre conceito e prática de casa item.

Figura 22 – Os fatores para a implementação ágil na indústria automotiva.



Fonte: Vasytkov (2019).

Com a inserção da abordagem ágil no desenvolvimento de projetos, o setor automotivo pode ser beneficiado em:

- Diminuição do tempo de colocação no mercado;
- Aumento da eficiência de custos;
- Criação de produtos com base nas necessidades dos clientes;
- Aumento da qualidade do produto;
- Lidar com a complexidade;
- Aumentar a capacidade de resposta.

EDAG Group (2019) relata em seu vídeo sobre o Scrum na engenharia automotiva que há convicção de que os requisitos em nosso tempo estão mudando muito rapidamente. Haverá um tempo que a comercialização será muito mais curta, e é por isso que com o uso do Scrum será possível fazer o dobro do trabalho pela metade do tempo. E também é possível notar maior produtividade e menores custos de desenvolvimento, porque é concentrado no que realmente importa. Isso leva a maior produtividade e menores custos de desenvolvimento, pois o foco está nos requisitos de maior valor. O benefício é um tempo de comercialização muito menor e uma satisfação muito melhor das necessidades reais dos clientes. Como resultado, menos projetos com falha são esperados e os riscos do projeto podem ser melhor controla-

dos. E no ambiente de desenvolvimento automotivo ou de hardware, é difícil construir um protótipo de *hardware* apresentável no final de cada *Sprint*, porque existem altos custos sobre a realização de protótipos, entre tanto, a motivação de usar o *framework* Scrum para desenvolvimento de *hardware* é positivo quando os desafios são superados, os benefícios são principalmente a aceleração da execução dos projetos, maior produtividade e menores custos de desenvolvimento, porque é focado no que realmente é importante. Os requisitos do produto também atentam as necessidades reais do cliente de forma eficaz, resultando em um maior valor de mercado, assim sendo, esperado menos projetos justos e riscos de projeto reduzidos.

Wald (2019) define que a principal diferença entre o desenvolvimento de *software* e *hardware* é que para produtos de *hardware* a duração de uma *Sprint* geralmente não é suficiente para garantir o fornecimento de um incremento. Com a falta desses incrementos, não será possível adquirir nenhum *feedback* e, assim, o princípio de utilização do *framework* Scrum falha. Com isso, é necessário definir metas razoáveis para cada *Sprint*, por exemplo, a realização de um esboço de CAD ou um protótipo de papelão pode ser incrementos valiosos. Portanto, o Scrum estrutura tarefas em componentes menores e menos complexos, os incrementos.

Scrum tem sido usado para desenvolver *software*, *hardware*, *software* embarcado, redes de funções interativas, veículos autônomos, escolas, governo, *marketing*, gerenciar a operação da organização e quase tudo que usamos em nosso dia-dia nas nossas vidas, como indivíduos e sociedades. (SCHWABER; SUTHERLAND, 2017, p. 4).

3.3.1 Exemplo de aplicação do framework Scrum no segmento automotivo

Vasytkov (2019) define que a Tesla é um exemplo de implementação bem sucedida do *framework* Scrum no segmento automotivo. Além disso, a Tesla decidiu não agir como qualquer outra montadora, e optou em agir como uma empresa desenvolvedora de *software*. Lançando melhorias de acordo em que recebem o *feedback* de seus *stakeholders*, incentivando novas ideias, solucionando problemas no estágio de desenvolvimento e buscando a melhoria contínua. Assim, tanto a programação extrema quanto o Scrum ajudaram a Tesla a construir sua infraestrutura em torno da aceitação de mudanças, e trazendo constantemente inovações ao mercado consumidor. Além de tudo, a Tesla optou em recusar seguir os longos ciclos de desenvolvimento que para muitos são considerados padrão no setor automotivo. E essa abordagem iterativa deve se tornar um exemplo para os fabricantes de automó-

veis e incentivá-los a mudar sua mentalidade tradicional de desenvolvimento de *software* no Modelo em V.

Por exemplo, se os engenheiros da Tesla projetarem uma porta melhor, mais leve, mais segura e mais barata a Tesla poderá reprogramar sua fábrica para construir a nova porta e integrá-la ao processo de fábrica existente. A fábrica não foi codificada para construir apenas um carro. Em vez disso, é capaz de mudar e se ajustar à medida que novos requisitos ou *user stories* são adicionados e projetados para o *front end*. (FIELD, 2018, p.1).

Field (2018) define que os ciclos iterativos de desenvolvimento de *software* moldou a cultura da Tesla. Rejeitando seguir os longos ciclos de desenvolvimento considerado padrão no segmento automotivo há décadas. A Tesla não espera para lançar melhorias. Ao invés disso, utiliza os princípios do desenvolvimento ágil em seus processos de *design* e fabricação automotiva. Recebendo *feedback* regularmente, como se enriquecesse em novas ideias, resolvendo problemas, melhoria contínua e iteração. O uso do Scrum ajudou a Tesla a aperfeiçoar seus veículos e trazer inovações ao mercado consumidor que levariam anos para as empresas automotivas tradicionais lançassem no mercado.

Denner (2015) define que a Bosch utiliza o *framework* Scrum no processo de desenvolvimento de *software* há muitos anos e também começaram a usar em vários projetos de IoT (*Internet of Things*). Com esse *framework* é possível atender aos novos requisitos dos clientes e mercado cada vez melhor. O desenvolvimento ágil faz sentido se as tecnologias ou soluções para o desenvolvimento ainda não estiverem claras desde o início e se os requisitos para um novo produto mudarem ao longo do tempo.

O desenvolvimento ágil também ajudou em nossa colaboração com a Tesla. Fornecemos chassis e sistemas de segurança para seus veículos elétricos. Muitos desses componentes de *hardware* e *software* podem ser correspondidos com precisão aos requisitos do respectivo veículo e ajustados às características de manuseio desejadas. Esta aplicação foi concluída com sucesso com a Tesla dentro de um curto período de tempo, e o projeto provou ser bem adequado ao uso de métodos de engenharia ágeis. (DENNER, 2015, p1).

Denning (2020) define que à medida que os carros se tornam "computadores sobre rodas", o gerenciamento ágil se espalha do desenvolvimento de *software* para toda a empresa. E realizou uma entrevista com a Anna Sandberg, diretora de melhoria contínua e mudança na criação de produtos da Volvo Cars em Gothenburg, Suécia. Perguntando a ela sobre sua experiência e o seu papel na Volvo Cars na aplicação do *framework* Scrum e de ferramentas ágeis:

A Volvo entendeu que os carros estavam se tornando "computadores sobre rodas". Precisávamos de métodos adequados a esse propósito. Estávamos tentando desenvolver o carro físico e depois adicionar o software mais tar-

de. Vimos que, para construir esses “computadores sobre rodas”, precisávamos desenvolver o software e o hardware ao mesmo tempo de forma integrada. Inicialmente, era difícil fazer as pessoas entenderem essa mudança. Ainda hoje, precisamos nos lembrar dessa necessidade. (SANDBERG, 2020).

Lufthansa *Industry Solutions* (2020) define que está em um processo desafiador de adequação que exige tempo para que possa implementar novos requisitos no prazo, e acredita que os fabricantes de automóveis precisam confiar em equipes de desenvolvimento ágil. Com o auxílio deles, novas demandas, métodos, produtos e funções podem ser implementadas dentro do prazo, mesmo os prazos sendo apertados. A Lufthansa *Industry Solutions* está usando métodos ágeis para ajudar um fabricante a reconfigurar seu processo de coleta de dados de forma que seja possível calcular com precisão os valores de medição do novo *Worldwide Harmonized Light-Duty Vehicle Test Procedure* (WLTP) no futuro.

Os benefícios: O fabricante do carro pode aproveitar o suporte adicional para cumprir os novos regulamentos, continuando a se concentrar no seu negócio principal. Isso ocorre porque essa colaboração é baseada no Scrum, uma estrutura de gerenciamento de projetos. A equipe do projeto externo ágil assume o controle total da implementação do projeto. A equipe ágil é bem praticada e possui o conhecimento especializado correspondente da indústria automotiva. Além disso, seus métodos de trabalho estão em conformidade com a Lei Alemã de Emprego Temporário. O responsável pelo projeto do fabricante do carro está em contato próximo com o Scrum Master, que se relaciona com a equipe do projeto em nome do provedor de serviços. Com base nesse processo, os fabricantes de automóveis podem implementar seus projetos com rapidez, eficiência e pontualidade. (LUFTHANSA INDUSTRY SOLUTIONS, 2020, p.1).

Denning (2019) relata que a Toyota está em um processo de transição do mundo do Toyota *Production System* (TPS) para o *framework* Scrum. O Sistema de Produção Toyota visa os problemas como complicados, mas essencialmente lineares, sendo ordenados e previsíveis com uma atribuição fixa de recursos. O TPS também foi construído para ser trabalhado em equipe. Entretanto, a Toyota percebeu que apenas com membros que trabalham para cumprir a missão e os objetivos da equipe (*Team Processes*), não seria o suficiente. Assim, ingressando ao mundo do *framework* Scrum e enxergando que para ser uma equipe, os membros precisam trabalhar de forma interdependente e adaptativa. E as equipes precisam aprender como auto avaliar e autocorriger, ou seja, é sobre como as pessoas funcionam juntas, um sistema adaptativo complexo. Para facilitar isso, a Toyota está empenhada em criar uma cultura de aprendizado no trabalho e desenvolvimento contínuo. É uma mudança de um sistema fechado, em que os subsistemas são mantidos por um sistema mestre, para o mundo de “fluxo” em que os problemas são reconhecidos co-

mo complexos e não complexos, não ordenado e imprevisível, exigindo uma realocação dinâmica de recursos. Um sistema aberto, que se adapta e se auto-organiza continuamente em resposta à mudança.

Linders (2018) realizou uma cobertura em nome da InfoQ no evento eXperience Agile 2018 em Portugal. E entrevistou Nigel Thurlow, líder de agilidade no projeto Toyota *Connected*, falando sobre o assunto “O jeito Toyota para Scrum”. A InfoQ faz coberturas editoriais desde 2006 sendo uma comunidade profissional independente, focada na disseminação de inovação e conhecimento no desenvolvimento de *software*. Na entrevista concedida, Nigel Thurlow falou sobre diversos assuntos relacionados à introdução da utilização do *framework* Scrum e de ferramentas ágeis na Toyota. E comenta como a Toyota *Connected* está aplicando uso destas ferramentas ágeis e do Scrum:

Estamos definindo o significado de agilidade para a Toyota como empresa global. Pegamos o conhecimento de muitas indústrias e estamos devolvendo para a comunidade quando tentamos encontrar a sinergia entre o Sistema Toyota de Produção/Lean e o mundo Ágil. Por exemplo, recentemente lançamos um treinamento aberto chamado 'O jeito Toyota para Scrum', e depois do sucesso dos testes iniciais estamos planejando uma disponibilidade mais abrangente deste treinamento. Continuamos aprendendo coisas novas e evoluindo à medida que nosso entendimento sobre este mundo se aprofunda. (THURLOW, 2018).

EDAG (2017) define que no futuro não iremos dirigir os veículos e estaremos apenas sendo movidos. Nas discussões sobre o *design* sustentável do automóvel, há evidências crescentes de ideias concretos sendo implementados em novos veículos. A direção totalmente automatizada, por exemplo, já é uma possibilidade técnica, e está sendo aguardada a legislação necessária para ser introduzido. O Collectivio é um conceito visionário de mobilidade para o futuro. Reinterpretando o conceito de ônibus clássico para se tornar um veículo elétrico, autônomo e em rede. Assim, mostrando como o processo de engenharia pode melhorar no futuro. Na fase conceitual do novo conceito completamente inovador de mobilidade e veículo o *framework* Scrum é a abordagem ideal mostrando como deve tornar o processo de engenharia mais flexível, de modo a responder mais rapidamente às demandas em constante mudança feitas no automóvel. No estande da EDAG no salão IAA *Commercial Vehicles* 2017 foi transformado em um laboratório com desenvolvedores experientes de seis divisões técnicas que definiram a tarefa de realizar o desenvolvimento ao vivo do Collectivio usando o Scrum durante o *show*. Foi empregado para garantir que os especialistas da EDAG pudessem reagir à entrada do público com flexibilidade e espontaneidade. E

com isso, os visitantes puderam interagir e desempenhar um papel ativo na fase conceitual. No final do teste de campo de 12 dias, as equipes de desenvolvimento da EDAG expressaram sua crença no potencial desse método de desenvolvimento para o segmento automotivo. Na figura 23 mostra o *layout* do projeto *Collectivo*.

Figura 23 - Projeto *Collectivo*.



Fonte: EDAG (2017).

Wald (2019) define que a EDAG Group participou junto com a MAN Truck & Bus no desenvolvimento do projeto chamado CitE. É o conceito de um caminhão elétrico com capacidade de 15 toneladas, sendo controlado por um motor elétrico que permite uma autonomia de pelo menos 100 quilômetros, e projetado para ser utilizado no tráfego de cargas na cidade. Em apenas 18 meses foi possível transformar a ideia em um veículo de exibição pronto para o salão IAA *Commercial Vehicles* 2018. É a principal feira mundial de mobilidade, transporte e logística. A concretização desse projeto foi possível graças à cooperação ágil entre as equipes de MAN e da EDAG, com a utilização do Scrum foi determinado o tempo de cada *Sprint* em uma semana. No fim de cada *Sprint* é feita a apresentação dos resultados obtidos na *Sprint Review* para as partes interessadas e para os clientes, adquirindo o *feedback* que influenciou de forma determinante no processo de desenvolvimento. Estes foram os fatores decisivos para o sucesso do projeto e o destaque na feira. A figura 24 mostra o modelo do conceito do caminhão elétrico CitE.

Uma equipe multifuncional foi elaborada com base no mesmo, que, com o método ágil de Scrum, organizou seu trabalho de maneira flexível e dinâmi-

ca. Revisões com os participantes do projeto em intervalos regulares aceleraram significativamente os processos de decisão. Usando essa abordagem ágil, o MAN CitE poderia, portanto, ser implementado em um período de apenas 18 meses e finalmente apresentado pronto para dirigir no IAA 2018. (GREEN CAR CONGRESS, 2018).

Figura 24 - MAN Truck & Bus apresenta o conceito de caminhão elétrico CitE.



Fonte: *Green Car Congress* (2018).

3.4 Empresas que já usam / usavam o Scrum

Por meio deste modelo de gerenciamento o *framework* Scrum é uma ferramenta muito útil e utilizada na gestão de planejamento no desenvolvimento de *softwares*. Algumas empresas já aderiram o uso do *framework* em seus projetos e outras já usou em algum momento para aplicar e ao fim analisar a viabilidade em relação à filosofia de trabalho da empresa.

Scrum é simples e leve. O que reforça a flexibilidade e possibilidade de uso do Scrum em equipes com uma unidade extremamente integrada com cada membro desempenhando um papel bem definido. Composto por práticas como papéis, eventos (atividades básicas) e artefatos. Cada elemento do *framework* tem um propósito específico e é fundamental para seu uso e sucesso. Com ele, não se gera ou aplica nada que não será efetivamente útil e utilizado. Eliminando controles desnecessários, inadequados ou burocráticos. São os inúmeros casos de utilizar do *framework* para desenvolver *software*, *hardware*, *software* embarcado, redes de funções iterativas, veículos, escolas, governo, *marketing* e gerenciar operações. A concentração

do trabalho está na essência do processo de desenvolvimento de sistemas ou *softwares*, mas também está presente em quase tudo que é utilizado nas empresas e em nossas vidas. Ribeiro (2016) define que muitas empresas utilizam o *framework* Scrum como ferramenta ágil para desenvolver seus projetos, entre elas destacam-se: Xerox, HP, Nokia, Yahoo, Microsoft, Once, Google, Philips, Siemens, BBC, Capital One, First American Real Estate, Nielsen Media, HP, Borland, Globo, Abril, UOL, Locaweb, entre outras. O Scrum se tornou essencial para muitas empresas atualmente, porque não apenas facilita a definição de objetivos, como também ajuda a cumprir os prazos estabelecidos.

Vito (2020) define que a Rede Globo é uma das grandes redes de televisão aberta brasileira e utiliza esse *framework* desde 2007 no desenvolvimento de seu *site*. Entretanto, houve dificuldades na definição de prioridades e as necessidades da empresa. Com isso, precisou-se de tempo para que pudessem se adaptar ao *framework*. Na utilização do Scrum, precisam-se expressar claramente os itens do *Backlog* do produto e é de grande importância para garantir que o *Scrum Team* possa trabalhar e seguir desenvolvendo os itens do *Product Backlog* sem fugir das prioridades.

Linders (2017) realizou uma entrevista em nome da InfoQ no evento *Agile Summit Greence 2017*. E entrevistou Eik Thyrsted Brandsgård, diretor na agência interna de *marketing* da LEGO. Na entrevista concedida, Eik Thyrsted Brandsgård falou sobre alguns assuntos relacionados ao uso de ferramentas ágeis na LEGO há mais de uma década. E comentou que quando entrou na LEGO em 2005 notou que ferramentas ágeis já estavam presentes com o uso do XP em pequena escala. O conhecimento ágil foi lentamente implementado e ganhou forças no ano de 2009, foi quando as equipes que desenvolveram o LEGO *Universe* foram certificadas em Scrum pelo Ken Rubin. Após o término do projeto algumas das equipes continuaram trabalhando muito bem com Scrum. Essa cultura foi mantida, recrutando e treinando pessoas em ferramenta ágil, mas como a LEGO cresceu muito acabou gerando alguns problemas com crescimento. Nos anos seguintes, foi ajustado o método e continuaram a identificar e remover interdependências, organizando a reduzir um pouco da complexidade e limitações. Por fim, o Scrum novamente foi escolhido como base.

FcAMERA (2017) O Yahoo! É um *portal web* de pesquisa que buscou introduzir o *framework* para otimizar tempo gasto em desenvolvimento de *software* enquan-

to gerencia o tamanho da equipe, com isso sendo possível ditar o fluxo de controle de aplicações, ou seja, ordenar os itens do *Product Backlog* para alcançar melhor as metas e missões otimizando o valor do trabalho que o Time de Desenvolvimento realiza.

Um dos maiores efeitos do Scrum dentro do Yahoo é como ele tem um impacto em manter a equipe unida. A empresa percebeu que grande parte dos desenvolvedores tinha uma tendência a trabalhar de forma isolada. Com uma maioria bem introvertida, eles tinham certa dificuldade em interagir e colaborar uns com os outros. Por isso, o Scrum no Yahoo teve um efeito um pouco diferente do que é visto nas outras empresas. Enquanto é comum observar os efeitos do Scrum em relação aos processos mais fluidos, por exemplo, no Yahoo foi percebida uma enorme diferença na forma como as pessoas interagem. (IEEP, 2020, p. 1).

Vito (2020) O Google desenvolve vários serviços e produtos com base na internet, e diversos setores escolheram utilizar *frameworks* ágeis para o desenvolvimento de *softwares*, assim, sendo possível criar e testar serviços ou produtos em pequenos intervalos de engenharia. Por serem auto-organizados, as equipes escolhem a tecnologia, as ferramentas que melhor podem ser aplicados e profissionais que possuem todas as habilidades necessárias para a resolução do problema. AdWords foi projetos em que se utilizou o Scrum em seu desenvolvimento e hoje em dia é gerado muito lucro com publicidade.

3.4.1 O modelo Spotify Squads

Mello (2018) define que o Spotify foi fundado na capital sueca Estocolmo, e atua no segmento de *streaming* de música. A empresa tornou-se um *case* de sucesso na organização de seus times orientados para produto e sendo reconhecido até por outras organizações. As equipes de desenvolvimento do Spotify tiveram grande influência de práticas ágeis, por exemplo, a utilização do *framework* Scrum e de conceitos de autogestão.

Nanga (2019) define que o Spotify começou a funcionar com a utilização do Scrum, mas quando começou a expandir suas atividades e número de equipes de desenvolvimento a estrutura deixou de ser eficiente. Com isso, deixou-se de usar o Scrum para começar a desenvolver um conceito próprio que seja mais adequado aos princípios da empresa. Com base no Scrum o Spotify criou uma nova abordagem com a seguinte estrutura:

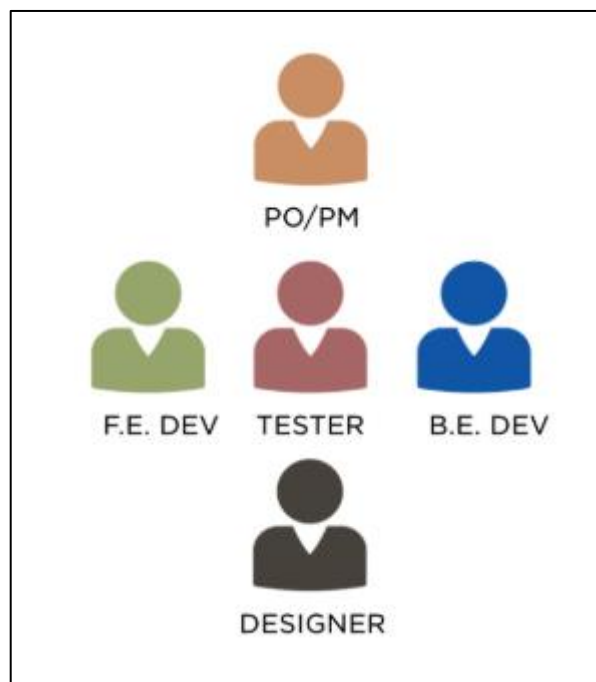
- O Scrum *Master* foi substituído pelo *Agile Coach*;

- O *Scrum Team* foi substituído pelo *Squad* (Esquadrão);
- Criação de novos grupos (*Tribes*, *Chapters* e *Guilds*).

Silva (2018) define que o Spotify alterou o nome do antigo *Scrum Master* para *Agile Coach* com o intuito de fazer com que seja menos um especialista em Scrum e se torne mais um líder servidor que seja capaz de estimular e suportar a melhoria contínua. E outra mudança foi alterar o nome das equipes multidisciplinares do *Scrum Team* para *Squads*.

Rebello (2013) define que os *Squads* são equipes semelhantes ao *Scrum Team*, à equipe é auto-organizável e independente para decidir seu próprio processo interno. Além disso, todas as equipes devem conter expertise dentro do grupo para desenvolver todos os aspectos do produto e cada equipe é responsável por desenvolver diferentes partes da experiência do usuário. Na figura 25 mostra a organização do *Squad*.

Figura 25 - Exemplo de organização do *Squad*.

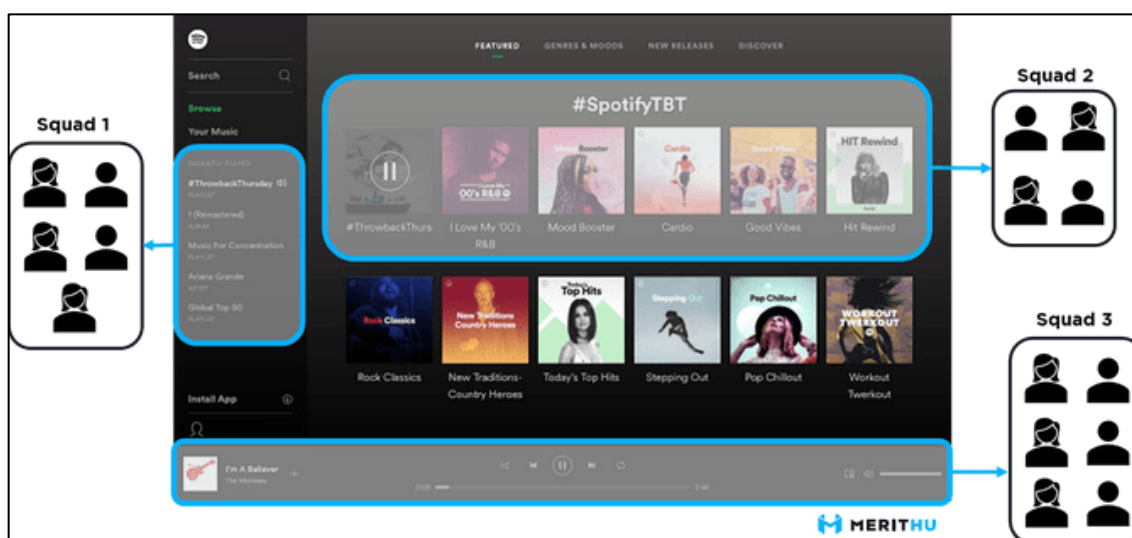


Fonte: Mello (2018).

Martini (2019) define que a equipe *Squad* é composta de três a dez pessoas com o foco em desenvolver uma funcionalidade específica. E o *Product Owner* da *Squad* determina quais são as prioridades de execução. Todas

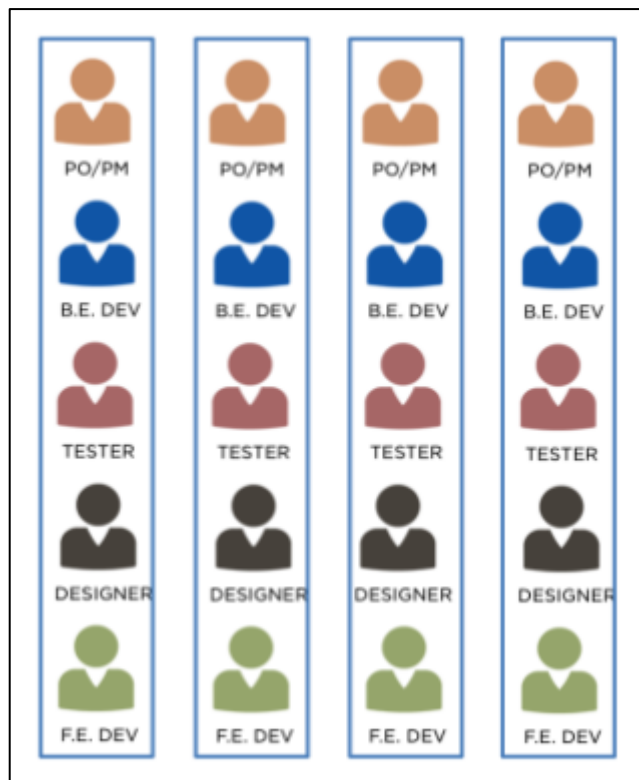
as *Squads* trabalham como se fossem pequenas *startup*, porque todas possuem uma missão, uma estratégia geral de projeto e objetivos de curto prazo, que são negociados a cada trimestre. Dentro dos *Squads* é utilizado princípios de *Lean Startup*, como o Mínimo Produto Viável (MVP) e Aprendizagem Validada. Na figura 26 representa a composição dos *Squads* em desenvolver funcionalidades específicas da plataforma.

Figura 26 – Composição dos *Squads* em desenvolver funcionalidades específicas.



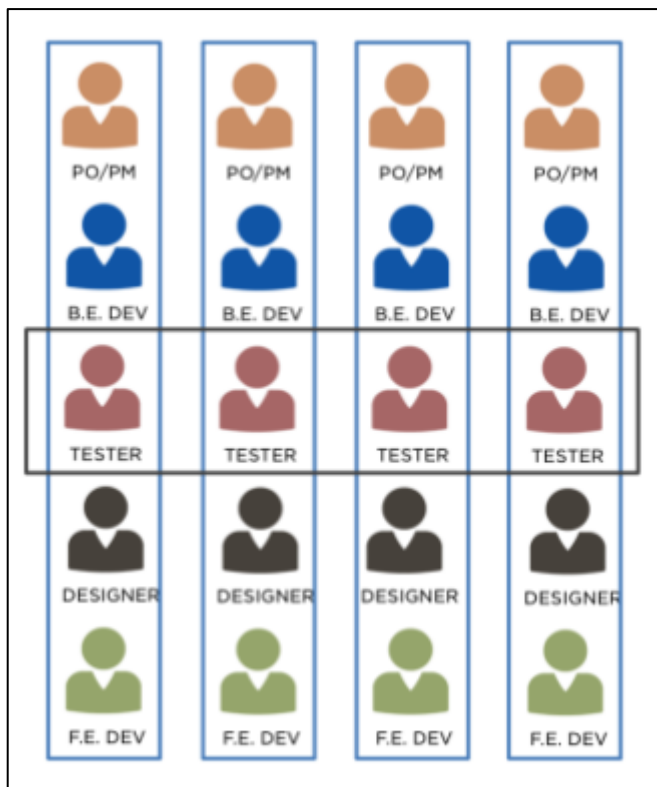
Fonte: Merithu (2019).

Mello (2018) define que a Tribo (*Tribes*) definido no Brasil como Tribo é o segundo nível de agrupamento e pode conter diversos *Squads* que tenham funções e objetivos em comum. Esses *Squads* ficam próximos para que tenha condições de comunicarem-se uns com os outros. Na figura 27 mostra a organização de vários *Squads* em uma *Tribe*.

Figura 27 - Exemplo de vários *Squads* de uma *Tribe*.

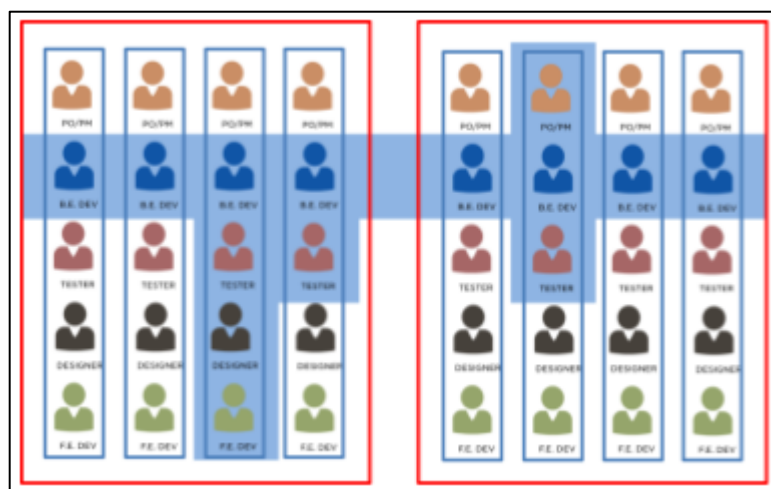
Fonte: Mello (2018).

Nanga (2019) define que o Capítulo (*Chapter*) pode ser visto como um “grupo de habilidades” que visa reunir profissionais de diferentes *Squads* que atuam no mesmo segmento para que possam trocar experiências, conhecimento e melhores práticas. Na figura 28 mostra a organização de *Tribe* que forma um *Chapter*.

Figura 28 - Exemplo de *Tribe* que forma um *Chapter*.

Fonte: Mello (2018).

Rebelo (2013) define que Aliança (*Guild*) são comunidades de interesse que visão em compartilhar conhecimento em um domínio específico, por exemplo, ferramentas, códigos e práticas. Enquanto os capítulos são locais a uma Tribo, as associações possuem integrantes provenientes de outras Tribos. Alguns exemplos de associações são associação de tecnologia *web*, associação de testes e associação de *Coaches* ágeis. Por exemplo, *Guildas* sobre liderança, desenvolvimento da *web*, entrega contínua. Na figura 29 mostra a organização de *Guilds* entre *Tribes*.

Figura 29 - Exemplo de *Guilds* entre *Tribes*.

Fonte: Mello (2018).

Martini (2019) define hoje em dia, o Spotify se tornou a principal plataforma de *streaming* de música do mundo, no ano de 2019 foi alcançada a marca de 217 milhões de usuários ativos. Atualmente, o Spotify possui mais de 50 *Squads* espalhados em quatro países, com o objetivo de desenvolver soluções inovadoras e realizar melhorias constantemente na experiência dos usuários dentro da plataforma.

4 CONCLUSÃO

Baseado no desenvolvimento teórico e nos resultados obtidos do presente trabalho, podemos concluir que a grande parte dos objetivos estabelecidos foi alcançada no início da elaboração do projeto e as demais foram concluídas de forma gradativas com algumas ressalvas. Devido a fatalidade da pandemia do Corona Vírus (COVID-19), Infelizmente não foi possível realizar atividades de levantamento de campo técnico em nível qualitativo relacionado ao uso do *framework* Scrum nas empresas que acreditam que investir em ferramentas de gestão de projetos são fontes potenciais de melhora no gerenciamento do desenvolvimento de projetos. Além disso, investigar e avaliar a viabilidade relacionada ao o uso do *framework* Scrum em termos financeiros, e apurar se houve aumentos na qualidade do produto, na satisfação dos clientes e, conseqüentemente, na redução de problemas de campo. Entre tanto, ao fim do trabalho, pode-se concluir que a pesquisa realizada ampliou o conhecimento a respeito do Scrum e forneceu informações importantes para entendermos a utilização de ferramentas ágeis na gestão de gerenciamento de projetos.

O gerenciamento ágil surge para suprir essa necessidade de gerenciar projetos sem burocracia, permitindo desenvolver com qualidade e de forma rápida para atender a demanda do mercado consumidor. O Scrum é um *framework* que se sobressaiu aos demais métodos ágeis que possuem estruturas lineares que os tornam rígidos, engessados e por serem incapazes de se adaptar a alterações durante o período de desenvolvimento. A estrutura incremental e eficiência dos resultados torna o Scrum compatível para qualquer segmento, mas foi no desenvolvimento de *software* que sua aceitação aumenta cada dia mais e o tornando cada vez mais popular no mundo. Entre tanto, o Scrum também está sendo capaz de ser aplicado no desenvolvimento de *hardware*. A índole ágil do *Scrum* o torna naturalmente compatível com a complexidade dos projetos e a necessidade de planejar conforme o desenvolvimento caminha.

Percebe-se que, de acordo com o uso do Scrum nota-se que é possível implementar algumas partes do Scrum no desenvolvimento de algum projeto, mas entretanto o resultado não será propriamente Scrum. O *framework* Scrum existe somente em sua totalidade e funciona muito bem como base e utilizar técnicas, métodos ágeis, práticas ou até mesmo o uso de normas técnicas para complementar o processo de desenvolvimento. Ou seja, tudo depende de como será implementado.

O Scrum é um processo que se bem utilizado e estruturado pode agregar um grande valor ao produto final. Vale ressaltar que o Scrum é uma ferramenta leve, simples de entender e difícil de dominar, com isso, precisa levar em consideração que ter uma boa equipe que tenha comprometimento em colaborar na melhoria constante do processo e com a prática levará a melhoria contínua. Também foi possível aprender que é de grande ajuda para quaisquer empresas que deseja inserir métodos ágeis em seus processos de desenvolvimento de projetos é focar em estimular as equipes a apreender como auto avaliar e autocorriger. Sobre como as pessoas podem funcionar juntas e tornando-se um sistema adaptativo complexo. Criando uma cultura de aprendizado no trabalho e desenvolvimento contínuo. Enfim, a adoção da utilização de ferramentas ágeis devem auxiliar a execução dos processos de gerenciamento e trazer maior confiabilidade.

Como proposta para trabalhos futuros ficam alguns tópicos como, por exemplo, customização de um modelo de gerenciamento que utilize a estrutura do Scrum para investigar e avaliar os ganhos relacionados à qualidade do produto, a redução de problemas de campo, a satisfação do cliente, dentre outras variáveis que poderão ocorrer. Os procedimentos de como aplicar o *framework* Scrum no gerenciamento de projetos automotivos na Indústria 4.0, apresentarem quais seriam as dificuldades para desenvolver essa aplicação em um país economicamente emergente que tem a necessidade de mostrar para o seu mercado consumidor interno e externo que há potencial para desenvolver grandes projetos e propor uma possível solução de como envolver todos os setores produtivos da empresa, podendo ser aplicado em qualquer área de atuação em que existam pessoas, as quais precisam trabalhar em conjunto para obter um objetivo comum.

5 REFERÊNCIAS BIBLIOGRÁFICAS

ABREU, Cleyde Evangelista Maia. **Indústria 4.0: Como as empresas estão utilizando a simulação para se preparar para o futuro**. Disponível em: <http://scholar.google.com.br/scholar_url?url=https://revista.pgsskroton.com/index.php/rcext/article/download/5444/4255&hl=pt-BR&sa=X&scisig=AAGBfm3nUTz_AK5e_X2Bl8pfnd67Fy9VOQ&nossl=1&oi=scholar>. Acessado em: 25 Jan. 2020.

ALCÂNTRA, Matheus Felipe Silva de. **A Indústria 4.0 introduzida na Alemanha aplicada no Brasil**. Disponível em: <<http://anais.unievangelica.edu.br/index.php/CIPEEX/article/download/2216/1668/>>. Acessado em: 16 Mai. 2020.

ALMEIDA, Julio Sergio Gomes de. **A indústria do futuro no Brasil e no mundo**. Disponível em: <https://iedi.org.br/media/site/artigos/20190311_industria_do_futuro_no_brasil_e_no_mundo.pdf>. Acessado em: 26 Jan. 2020.

ALTUS. **Conheça os nove pilares da indústria 4.0 e sua relevância para a atividade industrial**. Disponível em: <<https://www.altus.com.br/post/212/conheca-os-nove-pilares-da-industria-4-0-e-sua-relevancia-para-a-atividade-industrial>>. Acessado em: 26 jan. 2020.

ANTONIAZZI, Felipe. **Gestão de riscos cibernéticos**. Disponível em: <<https://blog.nec.com.br/gestao-de-riscos-ciberneticos>>. Acessado em: 19 jan. 2020.

ARKTIS. **Indústria 4.0, a Quarta Revolução Industrial**. Disponível em: <<http://arktis.com.br/a-quarta-revolucao-da-industria/>>. Acessado em: 26 Jan. 2020.

BATHIA, Sidharth. SPS (Scaled Professional Scrum): Quick Reference Guide & Exam Questions. 1. Ed. Sidharth Bathia: 2019.

BERNARDO, Kleber. **O que são métodos ágeis?**. Disponível em: <<https://www.culturaagil.com.br/o-que-sao-metodos-ageis/>>. Acessado em: 02 Fev. 2020.

BEZERRA, Juliana. **Revolução Industrial**. Disponível em: <<https://www.todamateria.com.br/revolucao-industrial/>>. Acessado em: 16 Mai. 2020.

BEZERRA, Filipe, **Stakeholder: Do significado á classificação**. Disponível em: <<https://www.portal-administracao.com/2014/07/stakeholders-significado-classificacao.html>>. Acessado em: 03 Mai. 2020.

BIANCHI, Isabella, **Minimum Viable Product (MVP): As vantagens de prototipar**. Disponível em: <https://econsult.org.br/blog/minimum-viable-product-mvp-as-vantagens-de-prototipar/?gclid=CjwKCAjwztL2BRATEiwAvnALcpi8TStQRjzDBSp09Clenr81ZcyYfZ0t4HNpfdHVvW-Spqqd6Q0qFBoCJOsQAvD_BwE>. Acessado em: 03 Mai. 2020.

BITTNER, Kurt, The Nexus framework for scaling Scrum. 1. ed. Addison-Wesley Professional, 2017.

BRASILEIRO, Roberto. **Métodos ágeis: O que é e porque você deve saber o que é.** Disponível em: <<http://www.metodoagil.com/metodos-ageis/>>. Acessado em: 02 Fev. 2020.

CAMARGO, Robson. **Metodologia ágil garante mais entrega de valor ao seu cliente.** Disponível em: <<https://robsoncamargo.com.br/blog/o-que-e-metodologia-agil>>. Acessado em: 02 Fev. 2020.

CAMPOS, Paola Souto. **Indústria 4.0: Um novo conceito de gerenciamento nas indústrias.** Disponível em: <https://semanaacademica.org.br/system/files/artigos/artigo_industria_4.0_-_revisao_em_29.11.2018.pdf>. Acessado em: 16 Mai. 2020.

CNI, Confederação Nacional da Indústria. **Indústria 4.0 e a digitalização da economia.** Disponível em: <https://bucket-gw-cni-static-cms-si.s3.amazonaws.com/media/filer_public/20/1c/201c2ab1-6f84-4190-8e71-533e334e5d86/32_-_industria_4_0.pdf>. Acessado em: 09 Fev. 2020.

COELHO, Pedro Miguel Nogueira. **Rumo à Indústria 4.0.** 2016. Disponível em: <<https://estudogeral.uc.pt/bitstream/10316/36992/1/Tese%20Pedro%20Coelho%20Rumo%20%c3%a0%20Industria%204.0.pdf>>. Acesso em: 19 jan. 2020.

CONTEZINI, Diego B., **4 vantagens e desvantagens do Scrum para negócios focados em Saas.** Disponível em: <<https://blog.asaas.com/4-vantagens-e-desvantagens-do-scrum-para-negocios-focados-em-saas/>>. Acessado em 19 Ju. 2020.

CRUZ, Fábio. Scrum e PMBOK® unidos no Gerenciamento de Projetos. 1. ed. Rio de Janeiro: Brasport, 2013.

D'AVILLAR, Priscila. **Você sabe o que são Frameworks? E para que servem?** Disponível em: <<https://dinamicatreinamentos.com/blog/o-que-sao-frameworks/>>. Acessado em: 23 Fev. 2020.

DENNER, Volkmar, **Agility at Bosch: mission impossible?** Disponível em: <<https://blog.bosch-si.com/digital-transformation/agility-at-bosch-mission-impossible/>>. Acessado em> 21 Mai. 2020.

DENNING, Steve, **Why and how Volvo Embraces agile at scale.** Disponível em: <<https://www.forbes.com/sites/stevedenning/2020/01/26/how-volvo-embraces-agile-at-scale/#65ce0e354cf0>>. Acessado em: 22 Mai. 2020.

DENNING, Steve, **World agility forum celebrates excellence, flays fake agile.** Disponível em: <<https://www.forbes.com/sites/stevedenning/2019/10/13/world-agility-forum-celebrates-excellence-attacks-fake-agile/#3e8de6408f84>>. Acessado em: 22 Mai. 2020.

DIAS, Ricardo Pereira, **O Modelo em Cascata**. Disponível em: <<https://medium.com/contexto-delimitado/o-modelo-em-cascata-f2418addaf36>>. Acessado em: 30 Mai. 2020.

DINAMIZE. **Scrum – O que é, suas etapas e como funciona na prática**. Disponível em: <<https://www.dinamize.com.br/blog/scrum/>>. Acessado em: 25 Fev. 2020.

DUARTE, Luiz, **9 vantagens de usar Scrum em seus projetos**. Disponível em: <<https://www.luiztools.com.br/post/9-vantagens-de-usar-scrum-em-seus-projetos/>>. Acessado em: 19 Jun. 2020.

DREHER, Andreas. **The Smart Factory of the Future**. Disponível em: <<https://www.belden.com/blog/industrial-ethernet/the-smart-factory-of-the-future>>. Acessado em: 16 Mai. 2020.

EDAG, **We develop the future EDAG annual report 2017**. Disponível em: <<https://ir.edag.com/download/companies/edag/Annual%20Reports/CH0303692047-JA-2017-EQ-E-00.pdf>>. Acessado em: 07 Mai. 2020.

EDAG, **Agiles projektmanagement**. Disponível em: <<https://www.edag.com/de/agiles-projektmanagement>>. Acessado em: 24 Mai. 2020.

EDAG GROUP, **Scrum in automotive engineering (English video)**. 2019. (05m08s). Disponível em: <<https://www.youtube.com/watch?v=HD8II3NZCYI>>. Acessado em: 24 Mai. 2020.

EXAME, **O Brasil está pronto para a indústria 4.0?**. Disponível em: <<https://exame.abril.com.br/tecnologia/o-brasil-esta-pronto-para-a-industria-4-0/>>. Acessado em: 26 Jan. 2020.

FCAMARA, **4 empresas que utilizam a metodologia Scrum em seus projetos**. Disponível em: <<https://blog.fcamara.com.br/4-empresas-que-utilizam-a-metodologia-scrum-em-seus-projetos/>>. Acessado em: 09 Jun. 2020.

FIELD, Kyle, **Tesla has applied agile software development to automotive manufacturing**. Disponível em: <<https://cleantechnica.com/2018/09/01/tesla-has-applied-agile-software-development-to-automotive-manufacturing/>>. Acessado em: 21 Mai. 2020.

FIRESMITH, Donald, **Using V Models for Testing**. Disponível em: <https://insights.sei.cmu.edu/sei_blog/2013/11/using-v-models-for-testing.html>. Acessado em: 19 Mai. 2020.

FRANCO, Eduardo Ferreira. **Um modelo de gerenciamento de projetos baseado nas metodologias ágeis de desenvolvimento de software e nos princípios da produção enxuta**. Disponível em: <https://www.teses.usp.br/teses/disponiveis/3/3141/tde-09012008-155823/publico/MS_Documento.pdf>. Acessado em: 01 Fev. 2020.

GARCIA, Pedro. **Robôs Autônomos**. Disponível em: <<https://pedrogarcia12av1.wordpress.com/about/robos-autonomos/>>. Acessado em: 25 jan. 2020.

GONÇALVES, Fabrício Moysés Polato. **Análise organizacional dos requisitos da indústria 4.0 com base em métodos multicritérios**. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/10117/1/CT_COEAU_2017_1_17.pdf>. Acessado em: 16 Mai. 2020.

GUEDES, Marylene. **Para que serve um *framework*?**. Disponível em: <<https://www.treinaweb.com.br/blog/para-que-serve-um-framework/>>. Acessado em: 23 Fev. 2020.

GIRALDO, Valentina. **Descubra finalmente o que é o cloud computing e para que serve a computação em nuvem**. Disponível em: <<https://rockcontent.com/blog/cloud-computing/>>. Acessado em: 19 jan. 2020.

GREEN CAR CONGRESS, **MAN Truck & Bus presents electric delivery truck CitE at IAA 2018**. Disponível em: <<https://www.greencarcongress.com/2018/09/20180920-cite.html>>. Acessado em: 07 Mai. 2020.

ICONFINDER, **Parceiros de negócios, investidores, acionistas**. Disponível em: <https://www.iconfinder.com/icons/2662137/business_partners_investors_shareholders_shareowners_stakeholders_icon>. Acessado em: 18 Mai. 2020.

IDEI, Instituto de Estudos para o Desenvolvimento Industrial. **Indústria 4.0 em perspectiva comparada**. Disponível em: <https://iedi.org.br/artigos/top/analise/analise_iedi_20180705_inovacao.html>. Acessado em: 02 Fev. 2020.

IDEI, Instituto de Estudos para o Desenvolvimento Industrial. **Indústria 4.0: A Política Industrial da Alemanha para o Futuro**. Disponível em: <https://iedi.org.br/cartas/carta_iedi_n_807.html>. Acessado em: 02 Fev. 2020.

IEEP, **Veja como e porque o Yahoo utiliza a metodologia Scrum**. Disponível em: <<https://www.ieepeducacao.com.br/yahoo-scrum/>>. Acessado em: 04 Mai. 2020.

JUNIOR, Carlos. **Quais são os principais tipos de métodos ágeis**. Disponível em: <<https://www.projectbuilder.com.br/blog/quais-sao-os-principais-tipos-de-metodos-ageis/>>. Acessado em: 23 Fev. 2020.

KENDIS, **The Nexus framework by Scrum.org for Scaling Agile**. Disponível em: <https://medium.com/@media_75624/the-nexus-framework-by-scrum-org-for-scaling-agile-c4a82b94bbb8>. Acessado em: 21 Jun. 2020.

KUGLER MAAG CIE, **Agile in automotive 2015**. Disponível em: <<https://www.kuglermaag.com/de/improvement-concepts/agile-in-automotive/state-of-practice-2015.html>>. Acessado em: 24 Mai. 2020.

KUGLER MAAG CIE, **Disciplined yet flexible**. Disponível em: <<https://www.kuglermaag.com/improvement-concepts/agile-in-automotive.html>>. Acessado em: 24 Mai. 2020.

LAMELAS, Ana, **As 5 principais metodologias agile: vantagens e desvantagens**. Disponível em: <<https://www.xpand-it.com/pt-pt/2018/10/11/5-metodologias-agile/>>. Acessado em: 21 Jun. 2020.

LAYTON, Mark C., David Morrow. **Scrum para leigos**. 2 ed. Rio de Janeiro: Alta Books, 2019.

LOPES, Luísa dos Prazeres, **Aplicação da metodologia Scrum em uma área da engenharia de processos de uma empresa do varejo**. Disponível em: <<http://monografias.poli.ufrj.br/monografias/monopoli10020157.pdf>>. Acessado em: 17 Jun. 2020.

LINDERS, Ben, **O ágil na LEGO**. Disponível em: <<https://www.infoq.com/br/news/2017/11/agile-lego/>>. Acessado em: 10 Jun. 2020.

LINDERS, Ben. **O jeito Toyota para Scrum**. Disponível em: <<https://www.infoq.com/br/articles/scrum-the-toyota-way/>>. Acessado em: 04 Mai. 2020.

LOTZ, Mary, **Waterfall vs. Agile: Which is the Right Development Methodology for your Project?**. Disponível em: <<https://www.seguetech.com/waterfall-vs-agile-methodology/>>. Acessado em: 30 Mai. 2020.

LUFTHANSA INDUSTRY SOLUTIONS, **Using agile methods to become more competitive**. Disponível em: <<https://www.lufthansa-industry-solutions.com/de-en/solutions-products/automotive/automotive-more-competitive-with-agile-methods/>>. Acessado em: 21 Mai. 2020.

MANCHADA, Amit. **Um guia passo a passo para criar um produto viável mínimo (MVP)**. Disponível em: <<https://www.netsolutions.com/insights/how-to-build-an-mvp-minimum-viable-product-a-step-by-step-guide/>>. Acessado em: 23 Fev. 2020.

MARTINI, Roberto, **Cultura ágil: O modelo Spotify Squads**. Disponível em: <<https://merithu.com.br/2019/09/18/cultura-agil-o-modelo-spotify-squads/>>. Acessado em: 07 Mai. 2020.

MARTINS, Jaqueline. **Metodologia ágil – Framework Scrum – Em gerenciamento de projetos de software**. Disponível em: <<https://revista.unilins.edu.br/index.php/cognitio/article/view/249/239>>. Acessado em: 01 Fev. 2020.

MARTINS, José Carlos Cordeiro. **Técnicas para gerenciamento de projetos de software**. 1. Ed. Rio de Janeiro: Brasport, 2007.

MASSARI, Vitor L. Agile Scrum Master no gerenciamento avançado de projetos: base para certificação EXIN Agile Scrum Master. 1. Ed. Rio de Janeiro: Brasport, 2016.

MAYR'S ORGANIZATION MANAGEMENT, **Stakeholders and Stakeholder Orietation**. Disponível em: <<https://mayrsom.com/2013/07/10/stakeholder-orientation/>>. Acessado em: 27 Jun. 2020.

MEIRELLES, Luiz Fernando T. **Framework para Aprendizagem com Mobilidade**. Disponível em: <<https://pdfs.semanticscholar.org/bc10/edc3d78f0cf94b4a532ef4d7530ffadc7fa7.pdf>>. Acessado em: 01 Fev. 2020.

MELO, Matheus Ribeiro. **Indústria 4.0 inovando com qualidade**. Disponível em: <<http://45.4.96.34/index.php/CIPEEX/article/view/2177/1661>>. Acessado em: 26 Jan. 2020.

MELLO, Francisco Homem de, **Como a Spotify organiza seus times de produto**. Disponível em: <<https://culture.rocks/blog/como-a-spotify-organiza-seus-times-de-produto/>>. Acessado em 31 Mai. 2020.

MORSE, Andrew Powell, **Waterfall Model: What is it and when should you use it?**. Disponível em: <<https://airbrake.io/blog/sdlc/waterfall-model>>. Acessado em: 30 Mai. 2020.

MORSE, Andrew Powell, **V-Model: Whats is it and How do you use it?**. Disponível em: <<https://airbrake.io/blog/sdlc/v-model>>. Acessado em: 04 Mai. 2020.

MIND MASTER, **Scrum: a metodologia ágil explicada de forma definitiva**. Disponível em: <<https://mindmaster.com.br/scrum/>>. Acessado em: 20 Jun. 2020.

NANGA, Melissa Manga, **Comparisons of agile flavors: Scrum, Safe, and Spotify**. Disponível em: <<https://www.initio.eu/blog/2019/3/7/comparisons-of-agile-flavors-scrum-safe-and-spotify>>. Acessado em: 31 Mai. 2020.

OPENCADD, **Engenharia de Requisitos no Setor Automotivo**. Disponível em: <<https://opencadd.com.br/engenharia-de-requisitos-no-setor-automotivo/>>. Acessado em: 19 Mai. 2020.

PAIOLA, Carlos. **4 aplicações na realidade aumentada (VA) na indústria 4.0**. Disponível em: <<https://www.itforum365.com.br/4-aplicacoes-de-realidade-aumentada-vr-na-industria-4-0/>>. Acessado em: 26 Jan. 2020.

PEREIRA, Leandro, **Nexus framework da Scrum.org, uma alternativa para escalar times Scrum**. Disponível em: <<https://agilebox.com.br/como-escalar-scrum-com-o-nexus-framework-da-scrum-org/>>. Acessado em: 21 Jun. 2020.

PORTAL DIRECTORY, **Non-Earth Observation mission**. Disponível em: <<https://directory.eoportal.org/web/eoportal/-/humsat>>. Acessado em: 19 Mai. 2020.

PIEIDADE, Antônio, **A perda da competitividade retira recursos ao nosso modelo social em causa muitas das conquistas civilizacionais**. Disponível em: <<https://www.sulinformacao.pt/2019/12/o-que-e-a-industria-4-0/>>. Acessado em: 29 Jul. 2020.

PILZ, **A produção do futuro – modular e aplicável**. 2020. Disponível em: <<https://www.pilz.com/pt-BR/products-solutions/industry-4-0/>>. Acesso em: 19 jan. 2020.

PMI, Project Management Institute. **Guia PMBOK®: Um Guia do Conhecimento em Gerenciamento de Projetos**. 6. ed. Newtown Square: Project Management Institute, 2017.

REBELO, Paulo, **Escalando o Agile na Spotify: exemplo de sucesso de Lean Startup, Scrum e Kanban**. Disponível em: <<https://www.infoq.com/br/articles/spotify-escalando-agile/>>. Acessado em: 31 Mai. 2020.

RIBEIRO, Celina Ferreira, **Como aplicar o Scrum em seus projetos**. Disponível em: <<https://www.devmedia.com.br/como-aplicar-o-scrum-em-seus-projetos/33996>>. Acessado em: 17 fev. 2020.

RIBEIRO, Matheus dos Santos. **A indústria 4.0 e a computação no Brasil**. Disponível em: <<http://repositorio.ufgd.edu.br/jspui/bitstream/prefix/2297/1/MatheusdosSantosRibeiro.pdf>>. Acessado em: 02. Fev. 2020.

RODRIGUES, Lucienne Keily da Silva. **Aplicação da metodologia ágil na gestão de um projeto de reservatório da metalúrgica carboquímica da Amazônia Ltda**. Disponível em: <http://www.abepro.org.br/biblioteca/TN_STO_238_378_34091.pdf>. Acessado em: 01 Fev. 2020.

ROQUE, Ruth Ferreira. **Estudo comparativo de metodologias de desenvolvimento de sistemas de informação utilizando a técnica Delphi**. Disponível em: <<https://repositorio.ufsc.br/xmlui/handle/123456789/77681>>. Acessado em: 02 Fev. 2020.

SABBAGH, Rafael. **Scrum: Gestão ágil para projetos de sucesso**. 1. ed. Brasil: Casa do Código, 2013.

SANTOS, Max Mauro Dias, **Indústria 4.0: Fundamentos, Perspectiva e Aplicações**. 1. ed. São Paulo: Érica, 2018.

SCHERMANN, Daniela, **Metodologia Scrum: como os métodos ágeis podem mudar a rotina da sua empresa**. Disponível em: <<https://blog.opinionbox.com/metodologia-scrum/>>. Acessado em: 21 Jun. 2020.

SCHWAB, Klaus. **A quarta revolução industrial**. São Paulo: Edipro, 2016.

SCHWABER, Ken. **Guia do Scrum – Um guia definitivo para o Scrum: As regras do jogo.** Disponível em: <<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Portuguese-Brazilian.pdf>>. Acessado em: 01 Fev. 2020.

SCHWABER, Ken, **Guia do Nexus™ - O guia definitivo para Nexus: O exoesqueleto do desenvolvimento Scrum escalado.** Disponível em: <<http://www.fabiocruz.com.br/livros/NexusGuide-v1.1-Versao-Portugues.pdf>>. Acessado em: 21 Jun. 2020.

STEFANI, Sergio. **Manufatura aditiva.** Disponível em: <<http://blog.render.com.br/diversos/manufatura-aditiva/>>. Acesso em: 26 Jan. 2020.

STRAFACCI, Gilberto, **A importância de metodologias ágeis para a Indústria 4.0.** Disponível em: <<https://www.setecnet.com.br/home/artigo-a-importancia-de-metodologias-ageis-para-a-industria-4-0-2/>>. Acessado em 20 Mai. 2020.

STRAFACCI, Gilberto, **Metodologia ágil é fundamental para avanço da Indústria 4.0.** Disponível em: <<https://cio.com.br/metodologia-agil-e-fundamental-para-avanco-da-industria-4-0/>>. Acessado em: 20 Mai. 2020.

SILVA, Edson. Scrum e TFS: uma abordagem prática. 1. Ed. Rio de Janeiro: Brasport, 2017.

SILVA, Debora. **Indústria 4.0: Iniciativas e evolução ao redor do mundo.** Disponível em: <<https://www.logiquesistemas.com.br/blog/industria-4-0-ao-redor-do-mundo/>>. Acessado em: 26 Jan. 2020.

SILVA, Leila Maciel de Almeida. **Uma análise das metodologias ágeis FDD e Scrum sob a perspectiva do modelo de qualidade MPS.Br.** Disponível em: <https://www.researchgate.net/publication/267250889_Uma_analise_das_Metodologias_Ageis_FDD_e_Scrum_sob_a_Perspectiva_do_Modelo_de_Qualidade_MPSBR>. Acessado em: 02 Fev. 2020

SILVA, André Marques Mancilha. **Aplicação da metodologia ágil Scrum no gerenciamento de projetos automotivos.** Disponível em: <<http://monografias.poli.ufrj.br/monografias/monopoli10020157.pdf>>. Acessado em: 02 Fev. 2020.

SILVA, Davi Gabriel da, **Modelo Spotify Squads: O que é e como funciona.** Disponível em: <<https://targetteal.com/pt/blog/modelo-spotify-squads/>>. Acessado em: 01 Jun. 2020.

SILVA, Tiago Mendonça da. **Aplicação de métodos ágeis no desenvolvimento de software automotivo – conciliação ou substituição ao ciclo-v.** Disponível em <<https://www15.fgv.br/network/tcchandler.axd?TCCID=8988>> Acessado em: 19 Mai. 2020.

SÓ HISTÓRIA. **Resumo - Revolução Industrial**. 2009. Disponível em: <<https://www.sohistoria.com.br/resumos/revolucaoindustrial.php>>. Acessado em: 19 jan. 2020.

TELE, Vinícius Manhães. *Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade*. 2. Ed. São Paulo: Novatec, 2014.

TEXEIRA, Daniel Dinis. **DSDM – Dynamic Systems Development Methodology**. Disponível em: <https://www.researchgate.net/profile/Daniel_Teixeira2/publication/237612979_DSDM_-_Dynamic_Systems_Development_Methodology/links/55ae197c08aed9b7dcdb2a0e/DSDM-Dynamic-Systems-Development-Methodology.pdf>. Acessado em: 02 Fev. 2020.

TRINDADE, Alex, **Stakeholder**. Disponível em: <<https://administradores.com.br/artigos/stakeholder>>. Acessado em: 03 Mai. 2020.

VASCONCELOS, Alexandre M. L. **Mapeamento do modelo de melhoria de processo de software brasileiro (MPS.Br) para empresas que utilizam Extreme Programming (XP) como metodologia de desenvolvimento**. Disponível em: <https://www.researchgate.net/publication/238775482_Mapeamento_do_modelo_de_Melhoria_do_Processo_de_Software_Brasileiro_MPSBr_para_empresas_que_utilizam_Extreme_Programming_XP_como_metodologia_de_desenvolvimento>. Acessado em: 02 Fev. 2020.

VASYLKOVA, Oleksandr, **How the automotive industry can benefit from agile software development**. Disponível em: <<https://clutch.co/developers/resources/how-automotive-industry-benefit-agile-software-development>>. Acessado em: 20 Mai. 2020.

VENÂNCIO, André Luiz A. C. **Sistema de avaliação de maturidade industrial baseado-se nos conceitos da Indústria 4.0**. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/10110/1/CT_COEAU_2017_1_09.pdf>. Acessado em: 16 Mai. 2020.

VENTURELLI, Marcio. **A Quarta Revolução Industrial – Uma Visão da Automação Industrial**. Disponível em: <<https://marcioventurelli.com/2014/09/02/industria-4-0/>>. Acesso em: 19 jan. 2020.

VINAL, Victor, **Métodos ágeis: O que são, como implementar as 4 principais**. Disponível em: <<https://rockcontent.com/blog/metodologias-ageis/>>. Acessado em: 20 Mai. 2020.

VITALLI, Rogério. **Os pilares da indústria 4.0**. Disponível em: <<https://www.industria40.ind.br/artigo/16751-os-10-pilares-de-industria-40>>. Acessado em: 25 Jan. 2020.

WALD, Daniel, **Scrum in Hardware Projects**. Disponível em: <<https://www.trive.me/scrum-meets-automotive-3-3/>>. Acessado em: 07 Mai. 2020.

WONG, K.Y.; ASPINWALL, E. **Knowledge Management Implementation Frameworks: A Review**. *Knowledge and Process Management*, v.11(2), pp.93-104, 2004.