



Universidade Federal do Rio de Janeiro

Escola Politécnica

MBA em Governança, Projetos e Serviços de TI  
(MGPS)

**ESTUDO DE CASO – TRANSFORMAÇÃO ÁGIL NA TI DE UMA  
EMPRESA DE TELECOMUNICAÇÕES**

**MELHORIAS DO PROCESSO DE TESTE DE SOFTWARE  
UTILIZANDO *SAFe***

Autor:

---

Marcia Nascimento das Neves

Orientador:

---

Wagner Rodrigues Ribeiro, M. Sc.

Examinador:

---

Cláudio Luiz Latta de Souza, M. Sc.

Examinador:

---

Marcela Lima dos Santos, D. Sc.

Examinador:

---

Vinicius Drumond Gonzaga, M. Sc.

**Rio de Janeiro  
Julho/2021**

## Declaração de Autoria e de Direitos

Eu, **Marcia Nascimento das Neves** CPF 071.089.047-80, autor da monografia ***ESTUDO DE CASO – TRANSFORMAÇÃO ÁGIL NA TI DE UMA EMPRESA DE TELECOMUNICAÇÕES - MELHORIAS DO PROCESSO DE TESTE DE SOFTWARE UTILIZANDO SAFe***, subscrevo para os devidos fins, as seguintes informações:

1. O autor declara que o trabalho apresentado na defesa da monografia do curso de Pós-Graduação, Especialização MBA - Engenharia de Computação e Sistemas da Escola Politécnica da UFRJ é de sua autoria, sendo original em forma e conteúdo.
2. Excetua-se do item 1 eventuais transcrições de texto, figuras, tabelas, conceitos e ideias, que identifiquem claramente a fonte original, explicitando as autorizações obtidas dos respectivos proprietários, quando necessárias.
3. O autor permite que a UFRJ, por um prazo indeterminado, efetue em qualquer mídia de divulgação, a publicação do trabalho acadêmico em sua totalidade, ou em parte. Essa autorização não envolve ônus de qualquer natureza à UFRJ, ou aos seus representantes.
4. O autor declara, ainda, ter a capacidade jurídica para a prática do presente ato, assim como ter conhecimento do teor da presente Declaração, estando ciente das sanções e punições legais, no que tange a cópia parcial, ou total, de obra intelectual, o que se configura como violação do direito autoral previsto no Código Penal Brasileiro no art.184 e art.299, bem como na Lei 9.610.
5. O autor é o único responsável pelo conteúdo apresentado nos trabalhos acadêmicos publicados, não cabendo à UFRJ, aos seus representantes, ou ao (s) orientador (es), qualquer responsabilização/ indenização nesse sentido.
6. Por ser verdade, firmo a presente declaração.

Rio de Janeiro, 28 de julho de 2021.

MARCIA NASCIMENTO DAS NEVES  
Nome Completo

## **UNIVERSIDADE FEDERAL DO RIO DE JANEIRO**

Av. Athos da Silveira, 149 - Centro de Tecnologia, Bloco H, sala - 212,  
Cidade Universitária Rio de Janeiro – RJ - CEP 21949-900.

Este exemplar é de propriedade Escola Politécnica da Universidade Federal do Rio de Janeiro, que poderá incluí-lo em base de dados, armazenar em computador, microfilmear ou adotar qualquer forma de arquivamento.

Permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do (s) autor (es).

## **AGRADECIMENTO**

“Agradeço primeiramente a Deus e Nossa Senhora por me permitirem concluir este curso e terem me dado saúde física e mental durante este difícil período da Pandemia do COVID-19.

Agradeço a ajuda e disponibilidade do mestre Wagner Ribeiro, que desde o primeiro momento que demonstrei interesse que ele fosse meu orientador, me recebeu de braços abertos e com um profissionalismo impecável.

E por fim, agradeço a minha família e meu futuro marido Ricardo Macedo por todo apoio, incentivo e amor que me dedicam, é por gratidão a este amor que me motivo cada vez mais para crescer como pessoa e como profissional. ”

## RESUMO

Na busca de melhorias nos processos da área de tecnologia da informação de uma grande empresa do setor de telecomunicações, frequentemente são revisadas e aprimoradas metodologias com objetivo de melhorar a qualidade das entregas que são feitas para a área de negócio e conseqüentemente para o cliente final. Com seus processos ultrapassados em relação a metodologias praticadas atualmente por outras empresas do setor, foi realizada uma mudança de processo com a utilização do Scaled Agile Framework (*SAFe*), com algumas adaptações necessárias para a empresa em questão. Este trabalho apresenta um estudo de caso da implementação de agilidade na TI e terá como foco principal a Área de Testes. Essa área é responsável pelos testes integrados e testes de aceitação dos usuários (UAT). Uma mudança nos procedimentos e processos desta empresa, requer paciência e persistência, pois se trata principalmente de mudança de cultura. Hoje todo seu processo de desenvolvimento é terceirizado e regido por contratos firmados com seus fornecedores. Os principais desafios da transformação ágil é diminuir a burocracia dos processos atuais, melhoraria na comunicação das áreas envolvidas, mais sinergia entre colaboradores e as fábricas de software e redefinição dos papéis e responsabilidades focando na qualidade e agilidade das entregas dos projetos desenvolvidos. Durante este estudo de caso e pesquisas relacionadas ao *SAFe* e outros métodos ágeis, foram identificadas melhorias que podem ser realizadas para a implementação de forma mais clara e objetiva este modelo de trabalho.

Palavras-Chave:

(*SAFe*, Testes, Processos, Fábrica de Software, Transformação Ágil).

## ABSTRACT

In the search for improvements in the processes of the information technology area of a large company in the telecommunications sector, methodologies are often reviewed and improved in order to improve the quality of deliveries that are made to the business area and consequently to the end customer. With its processes exceeded in relation to methodologies currently practiced by other companies in the sector, a process change was made with the use of the Scaled Agile Framework (*SAFe*), with some necessary adaptations for the company in question. This paper presents a case study of the implementation of agility in IT and will focus on the Testing Area. This area is responsible for integrated testing and user acceptance testing (UAT). A change in the procedures and processes of this company requires patience and persistence, because it is mainly about culture change. Today its entire development process is outsourced and governed by contracts signed with its suppliers. The main challenges of Agile transformation is to reduce the bureaucracy of current processes, improve communication of the areas involved, more synergy between employees and software factories and redefinition of roles and responsibilities focusing on the quality and agility of deliveries of the projects developed. During this case study and research related to *SAFe* and other agile methods, improvements were identified that can be made for the implementation of this work model more clearly and objectively.

Keywords:

(*SAFe*, Testing, Processes, Software Factory, Agile Transformation).

## SIGLAS

<b>ART</b>	Agile Release Train
<b>BRP</b>	Big Room Planning
<b>CALMR</b>	Cultura, Automação, Fluxo Enxuto, Medição e Recuperação
<b>CCQ</b>	Centro de Controle de Qualidade
<b>CD</b>	Continuous Deployment
<b>CDG</b>	Comitê de Gestão
<b>CDP</b>	Continuous Delivery Pipeline
<b>CE</b>	Continuous Exploration
<b>CFD</b>	Cumulative Flow Diagram
<b>CI</b>	Continuous Integration
<b>CRM</b>	Customer Relationship Management
<b>CT</b>	Caso de Teste
<b>DevOps</b>	Developments and Operations
<b>GT</b>	Gestão Técnica
<b>MBA</b>	Master in Business Administration
<b>MVP</b>	Minimum Viable Product
<b>PDCA</b>	Plan, Do, Check, Act
<b>RH</b>	Recursos Humanos
<b>PI</b>	Program Increment
<b>PM</b>	Product Manager
<b>PMO</b>	Project Management Office
<b>PO</b>	Product Owner

<b>R&amp;P</b>	Refinamento e Planejamento
<b>RTE</b>	Release Train Engineer
<b>SAFe</b>	Scaled Agile Framework
<b>SM</b>	Scrum Master
<b>TDD</b>	Test Driven Design
<b>TI</b>	Tecnologia da Informação (quando estiver fazendo referência a área/diretoria)
<b>TI</b>	Teste Integrado (quando estiver fazendo referência a etapa do teste de software)
<b>TRG</b>	Teste de Regressão Geral
<b>TS</b>	Teste de Sistemas
<b>TU</b>	Teste Unitário
<b>UAT</b>	User Acceptance Test
<b>UI</b>	User Interface
<b>US</b>	User Story
<b>UX</b>	User Experience
<b>VP</b>	Vice-Presidência
<b>WIP</b>	Work in Process
<b>XP</b>	eXtreme Programming

## LISTA DE FIGURAS

Figura 2.1.1	As sete competências essenciais para empresa enxuta	07
Figura 2.1.2	Pipeline de Entrega Contínua do <i>SAFe</i>	08
Figura 2.1.3	Abordagem CALMR do <i>SAFe</i> para o DevOps	09
Figura 2.1.4	Exemplo de um quadro Kanban	11
Figura 2.1.5	Comparativo de mudança para entrega antecipada	13
Figura 2.1.6	Ciclos de PDCA (Plan, Do, Check, Act)	14
Figura 2.2.1	Quadro Kanban	19
Figura 2.3.1	Fluxo Scrum	23
Figura 2.3.2	Exemplo da Dinâmica que pode ser feita durante a Retrospectiva da Sprint	25
Figura 3.1	Estrutura Gerencial da Diretoria de TI da Empresa de Telecom	33
Figura 3.1.1	Visão Macro da Governança de Projetos Ágeis	35
Figura 3.1.2	Atividades Camada de Gestão – Originação	36
Figura 3.1.3	Atividades Camada de Gestão – Gestão Técnica	36
Figura 3.1.4	Atividades Camada de Gestão – Desenvolvimento	36
Figura 3.1.5	Exemplo da Composição de um Trem de Produto	38
Figura 3.1.6	Exemplo das Squads Cross	38
Figura 3.2.1	Principais atividades da Gerência de Qualidade e suas áreas	50
Figura 3.2.2	Estrutura da Área de Gestão de Testes	52
Figura 3.2.1.1	Eventos no Ciclo da Camada Técnica	54
Figura 3.2.1.2	Eventos no Ciclo da Camada de Desenvolvimento	54
Figura 4.1.1	Gráfico do Tempo de Entrega da Estratégia de Testes (Waterfall x Ágil)	57
Figura 4.1.2	Gráfico do Tempo de Aprovação da Estratégia de Testes (Waterfall x Ágil)	57

Figura 4.1.3	Gráfico do Tempo de Planejamento dos Casos de Testes (Waterfall x Ágil)	58
Figura 4.1.4	Gráfico do Tempo de Aprovação dos Casos de Testes (Waterfall x Ágil)	58
Figura 4.1.5	Gráfico do Tempo para Aprovação de Proposta Financeira (Waterfall)	60
Figura 4.2.3.1	Macro Fluxo do Processo de Testes	63

## **LISTA DE TABELAS**

Tabela 4.1.1 Tempo Médio de Planejamento dos Testes

59

## LISTA DE QUADROS

Quadro 3.1.1	Atividades e Participantes dos Eventos e Atividades	37
Quadro 3.1.1.1	Papéis e Responsabilidades	39
Quadro 3.1.2.1	Ritos e Cerimônias – Camada de Originação	42
Quadro 3.1.2.2	Ritos e Cerimônias – Camada Técnica	43
Quadro 3.1.2.3	Ritos e Cerimônias – Camada Desenvolvimento	45

# Sumário

<b>Capítulo 1: Introdução</b> .....	<b>1</b>
1.1 – Tema .....	3
1.2 – Justificativa.....	3
1.3 – Objetivos.....	3
1.4 – Delimitação.....	4
1.5 – Metodologia.....	4
1.6 – Descrição .....	5
<b>Capítulo 2: Embasamento Teórico</b> .....	<b>6</b>
2.1 – O que é o <i>SAFe</i> ? .....	6
2.2 – Trabalhando com Kanban.....	18
2.2.1 – Princípios do Kanban.....	19
2.2.2 – Benefícios do Kanban.....	20
2.2.3 – As 6 Práticas do Método Kanban .....	21
2.2.4 – Cadeia Puxada com Gestão Visual.....	21
2.3 – Características do Scrum .....	22
2.4 – XP – Extreme Programming .....	27
2.5 – Conceitos de Testes de Software .....	29
<b>Capítulo 3: Capítulo 3: Estudo de Caso – Implantando a Transformação Ágil na Diretoria de TI</b> .....	<b>33</b>
3.1 – Transformação Ágil – Implantação do <i>SAFe</i> . .....	34
3.1.1 – Papéis e Responsabilidades. ....	39
3.1.2 – Ritos e Cerimônias.....	42
3.2. – Processo da Área de Gestão de Testes.....	49
3.2.1 – Resumo de Eventos com Insumos para Área de Testes .....	53
<b>Capítulo 4: Resultados Obtidos</b> .....	<b>56</b>
4.1 – Análise de Tempo de Entrega no Planejamento de Testes (Waterfall x Ágil) .....	56
4.2 – Análise sobre o Estudo de Caso e o Processo da Área de Testes.....	60
4.2.1 – Proposta de Ações de Melhorias na Área de Testes.....	61
4.2.2 – Revisão dos Papéis e Responsabilidades do Time de Qualidade .....	62
4.2.3 – Macro Fluxo do Processo de Gestão de Testes .....	63
4.2.4 – Outras Iniciativas para o Sucesso na Implantação do Processo .....	64
<b>Capítulo 5: Conclusão e Trabalhos Futuros</b> .....	<b>66</b>
5.1 – Conclusão .....	66
5.2 – Trabalhos Futuros .....	67
<b>Referências Bibliográficas</b> .....	<b>68</b>

# Capítulo 1

## Introdução

A área de Tecnologia da Informação (TI) de uma grande empresa de Telecomunicações vem atuando constantemente em melhorias dos processos de desenvolvimento, objetivando estruturar processos que possam modificar a forma de atuação de TI e viabilizar a atuação integrada de seus colaboradores, visando obter os resultados esperados de forma eficaz e alinhada com o negócio da companhia.

Os processos de TI estão sempre em constante evolução e para acompanhar a necessidade de aumentar a capacidade de entrega de valor, foi realizada uma mudança nos seus processos para utilização do *SAFe* (Scaled Agile Framework), já que o *SAFe* funciona bem em equipes ágeis, sendo possível fazer a gestão de programas e de maneira ágil tratar portfólios de demandas organizacionais. Como times ágeis são por natureza pequenos para facilitar a comunicação, para criar grandes produtos é indicado utilizar técnicas de escalar o ágil para que seja possível sincronizar o trabalho de vários times que tem como objetivo a entrega de grandes produtos.

Este trabalho tem como objetivo realizar um Estudo de Caso sobre as mudanças que estão sendo implementadas na companhia e apresentar, através de estudos realizados para entendimento teórico sobre frameworks e métodos ágeis, uma proposta de como essas mudanças deveriam ser feitas. O foco principal deste trabalho é uma área específica da TI, que é a área de Gestão de Testes, que faz parte da Gerência de Qualidade. Por se tratar de uma área Cross, ou seja, é uma área que permeia todas as fases e é de grande relevância no ciclo de vida de um projeto, essa área passa por constantes desafios para que suas entregas sejam realizadas de forma rápida, com qualidade e menor custo.

Atualmente essa área é suportada por contratos firmados com fábrica de software contratada para trabalhar em formato de baseline mensal, ou seja, alocação de horas mensais para atuação nos projetos. Essas horas são consumidas através de propostas técnicas que são elaboradas a cada projeto de acordo com a necessidade, além disso o contrato com este fornecedor não permite que eles atuem em Squads multidisciplinares.

O atual fornecedor que realiza as atividades de testes, tem também contrato firmado como fornecedor de desenvolvimento em outras áreas da TI. Como existem outros fornecedores

de desenvolvimento, que são responsáveis por outros sistemas, existe uma cláusula contratual onde diz que eles não atuam em Squads com fornecedores de desenvolvimento diferentes, isso gera o primeiro impasse em implantar uma metodologia ágil da forma correta.

Migrar para um processo ágil, requer mudança de cultura interna dos colaboradores e de toda liderança, como também mudanças no modelo de contratação dos serviços prestados pelas fábricas de software.

Além do exposto acima, a arquitetura sistêmica da companhia também é muito complexa, devido ao alto volume de sistemas de CRMs, Aprovisionamento, Faturamento, entre outros. Dependendo do serviço esses sistemas podem ser de responsabilidade de fornecedores e metodologias de trabalhos diferentes e a área de testes acaba sendo considerada como a área que impacta o cronograma dos projetos.

Será apresentado neste trabalho o processo desenhado para atuação no modelo ágil, com ênfase ao processo e atuação da área de testes e por fim, será apresentada uma proposta de mudança para que a área de testes possa atuar de forma eficaz, como: treinamento para os times envolvidos, engajamento das lideranças, revisão dos papéis e organização do time e proposta de revisão contratual.

## **1.1 – Tema**

Após o Estudo de Caso da Implantação da metodologia ágil na TI de uma grande empresa, entender de forma específica como a Área de Testes está atuando e, propor uma reestruturação no processo dessa área.

## **1.2 – Justificativa**

Uma organização deve ser capaz de se adaptar as mudanças e inovar seus processos para que seja adequado as novas realidades, porém para que as mudanças sejam feitas de forma efetiva é necessário rever os pontos onde se torna inviável a utilização dos métodos propostos e realizar os ajustes necessários para que a mudança seja realizada conforme a expectativa.

A maior motivação para o presente trabalho é reestruturar o processo da área de testes, utilizando métodos atuais para que as atividades desempenhadas por essa área sejam efetivas e reconhecidas pela sua importância.

## **1.3 – Objetivos**

### **Objetivo geral**

O objetivo deste trabalho é apresentar uma análise qualitativa de como está estruturado o processo ágil na diretoria de TI da companhia.

Anteriormente todo desenvolvimento era guiado pelo método tradicional Waterfall onde os projetos eram tratados de forma linear e as equipes envolvidas nos projetos só atuavam quando a etapa anterior finalizava suas atividades, com este método antigo existia muitas falhas de comunicação, altos custos e atrasos na entrega da solução.

Como as empresas estão sempre em busca de melhorias na sua operação para minimizar riscos e incertezas, foi identificada a necessidade de rever seus processos para que seja possível entregas mais rápidas, aumentar a percepção de qualidade dos clientes e reduzir a quantidade de coisas desenvolvidas e que no final acabam não sendo utilizadas devido ao tempo que demoravam para ficar prontas.

## **Objetivo específico**

Este trabalho irá apresentar como o processo ágil foi inicialmente definido na companhia. Quais as cerimônias definidas para apoiar o andamento dos projetos, quais papéis e responsabilidades foram estabelecidos tanto para os colaboradores da empresa quanto para seus fornecedores.

Uma das áreas que compõe a diretoria de TI é a Gerência de Qualidade, essa gerência é dividida nas áreas de Testes, Release, Garantia da Qualidade e Ambientes Não Produtivos.

Este trabalho irá abordar de forma específica como está estruturado o processo da Área de Testes e ao final apresentar uma proposta de melhorias.

### **1.4 – Delimitação**

Este trabalho irá focar no processo da área de Testes.

Não será abordado neste trabalho os processos das demais áreas da Gerência de Qualidade.

### **1.5 – Metodologia**

Para o desenvolvimento deste trabalho foi realizado um estudo de caso sobre a metodologia ágil que está sendo aplicada na empresa em questão e como a área de testes está atuando com a metodologia adotada.

Será apresentado o desenho atual do processo utilizando o framework *SAFe* e em destaque será apresentado como se dá a participação da área de testes neste contexto.

Através de estudos feitos para a realização deste trabalho será apresentado uma proposta para que a área de testes tenha melhor entendimento sobre o framework *SAFe* e demais métodos ágeis e possa atuar de forma colaborativa, engajando seus colaboradores e lideranças através de treinamentos e promovendo mais qualidade nas suas entregas.

Neste estudo não será possível apresentar uma abordagem mais ampla, considerando melhorias para o ciclo de vida como um todo, passando por todas as áreas, mas ficará como insumo para propostas futuras de melhoria nos processos da empresa.

## 1.6 – Descrição

Este trabalho está estruturado da seguinte forma:

No capítulo 2 será apresentado o embasamento teórico para este trabalho, onde será apresentado uma abordagem sobre o framework *SAFe*, Kanban, Scrum, Extreme Programming e Conceitos de Testes de Software.

O capítulo 3 apresenta o objeto do estudo de caso, de como foi feita a transformação ágil na empresa que é escopo deste trabalho, como foi implantado o framework *SAFe* e como a área de testes está atuando neste novo contexto, quais suas dificuldades na participação deste método de trabalho, e como é a atuação dos colaboradores desta área.

No capítulo 4, será apresentado os resultados obtidos com o estudo que foi realizado no Capítulo 3 e uma proposta de reestruturação da área de testes e sugestões de ações para que o time esteja engajado nas mudanças propostas.

E por fim, no capítulo 5 será apresentada a conclusão e sugestão de trabalhos futuros.

# Capítulo 2

## Embasamento Teórico

Este capítulo apresenta o embasamento teórico sobre os principais assuntos que serão abordados neste trabalho, nele será apresentada as principais competências do framework *SAFe*, orientações sobre o Kanban, Scrum, Extreme Programming e Conceitos de Testes de Software.

Abaixo um breve resumo dos pontos que serão descritos no decorrer deste capítulo:

- *SAFe* - orientação para trabalhar com portfólio, fluxo de valor, programa, equipe.
- Kanban - usa o conceito de cadeia puxada para expor problemas, gerar questionamentos e criar melhorias no fluxo de trabalho;
- Scrum - framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos;
- XP - eXtreme Programming, metodologia ágil que se ajusta bem em projetos que tenham requisitos vagos e em constante mudança;
- Teste de Software - etapa do ciclo de desenvolvimento de um produto, com controle de qualidade que envolve validações e verificações em várias etapas do processo.

### 2.1 – O que é o *SAFe*?

*SAFe* é um “conjunto de padrões de organização e fluxo de trabalho para implementar práticas ágeis em escala empresarial. A estrutura é um conjunto de conhecimentos que inclui orientação estruturada a respeito dos papéis e responsabilidades, como planejar e gerenciar o trabalho e os valores que devem ser defendidos” [1].

Entre outras coisas o *SAFe* permite a implementação de software Lean-Agile no nível corporativo, fornece orientação detalhada para trabalhar com portfólio, fluxo de valor, programa, equipe e atende às necessidades de todas as partes interessadas numa organização.

Em 2011 Dean Leffingwell e Drew Jemilo lançaram o *SAFe* para apoiar as empresas a desenvolver sistemas e produtos de software melhores e que atendessem melhor às necessidades em constante mudança dos clientes.

O uso do *SAFe* permite Agilidade no Negócio, através de sete competências essenciais, conforme mostra a figura 2.1.1. As descrições dessas competências serão apresentadas a seguir.

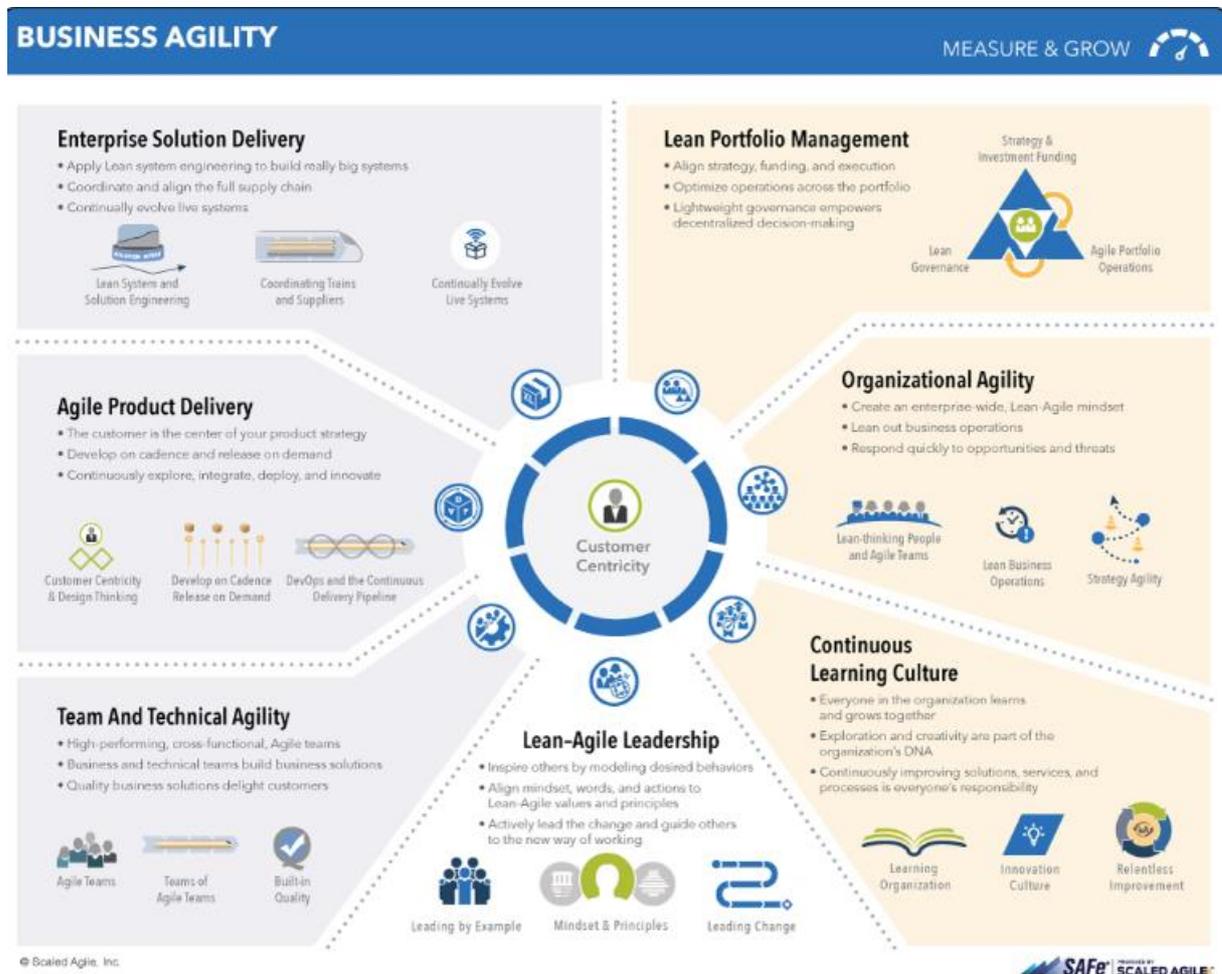


Figura 2.1.1 – As sete competências essenciais para empresa enxuta

Fonte: [2]

## 1 – Enterprise Solution Delivery (Entrega de Solução Empresarial) [3]:

A Entrega de Solução Empresarial descreve como aplicar os princípios e práticas Lean-Agile à especificação, desenvolvimento, implantação, operação e evolução nos maiores e mais sofisticados software, rede e sistemas do mundo. Construir e desenvolver grandes soluções gera um grande esforço e requer várias pessoas e áreas envolvidas.

Inovar requer experimentação e conhecimento em diversas disciplinas, ao longo do tempo os sistemas vão evoluindo e suas missões e propósitos mudam, exigindo novos recursos e atualizações tecnológicas.

O movimento DevOps tem práticas recomendadas avançadas para melhor suportar atualizações de sistemas frequentes por meio de “Continuous Delivery Pipeline (CDP)” que permite fornecer entregas mais rápidas e contínuas de valor. Os sistemas não são simplesmente implantados de uma vez, eles são lançados mais cedo e sendo aprimorados ao longo do tempo.

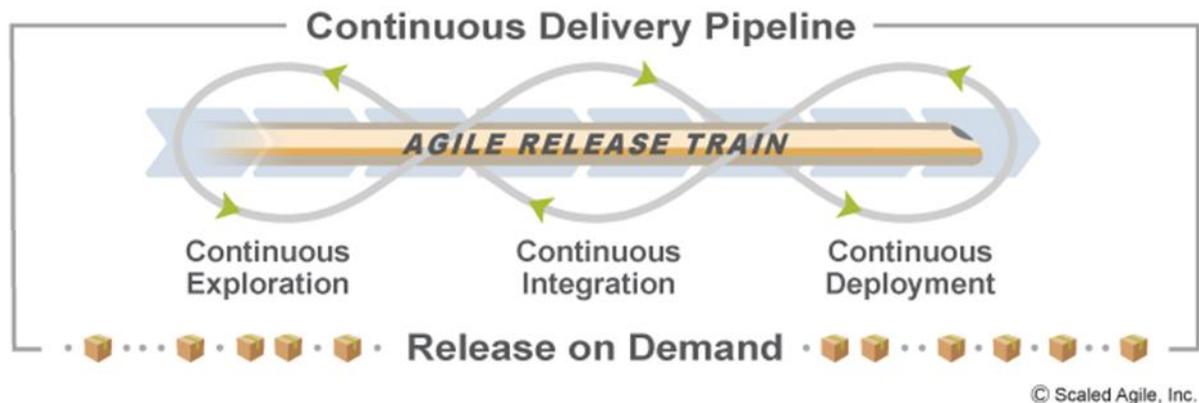


Figura 2.1.2 – Pipeline de Entrega Contínua do SAFe

Fonte: [4]

Como mostra a figura 2.1.2, o pipeline é um elemento significativo dessa competência. Cada Agile Release Train (ART) constrói e mantém um pipeline de ativos tecnológicos necessários para entregar novas funcionalidades aos usuários com muito mais frequência.

Os elementos CE (Continuous Exploration), CI (Continuous Integration) e CD (Continuous Deployment) trabalham juntos para oferecer suporte a entregas de pequenos lotes de novas funcionalidades para atender as demandas de mercado.

## 2 – Agile Product Delivery (Entrega Ágil de Produtos) [4]:

Para obter agilidade nos negócios, as empresas devem aumentar rapidamente sua capacidade de fornecer produtos e serviços inovadores, com foco no cliente. E são estabelecidas 3 dimensões para entrega ágil de produtos.

1 - Centralidade no Cliente e Design Thinking – Centralização no Cliente é assumir um compromisso positivo à medida que o cliente experimenta os produtos e serviços oferecidos. E colocá-lo a frente das decisões através do uso do design thinking, garantindo que a solução seja viável e sustentável durante todo o ciclo de vida.

2 - Desenvolver na Cadência; liberar sob demanda – É um conjunto coordenado de práticas que ajudam as equipes ágeis, fornecendo uma série de eventos e atividades que ocorrem

em um cronograma regular e previsível. Entregar valor garante que o cliente tenha o que precisa, quando precisa.

3 - DevOps e o Pipeline de entrega contínua – Criam a base para permitir que as empresas possam liberar valor para atender as necessidades dos clientes e do mercado. Criar a mentalidade de cultura DevOps é criar um canal de entrega continua cada vez mais automatizado. A abordagem *SAFe* para o DevOps é baseada em cinco conceitos: Cultura, Automação, Fluxo Enxuto, Medição e Recuperação (CALMR), como exemplificado na figura 2.1.3.

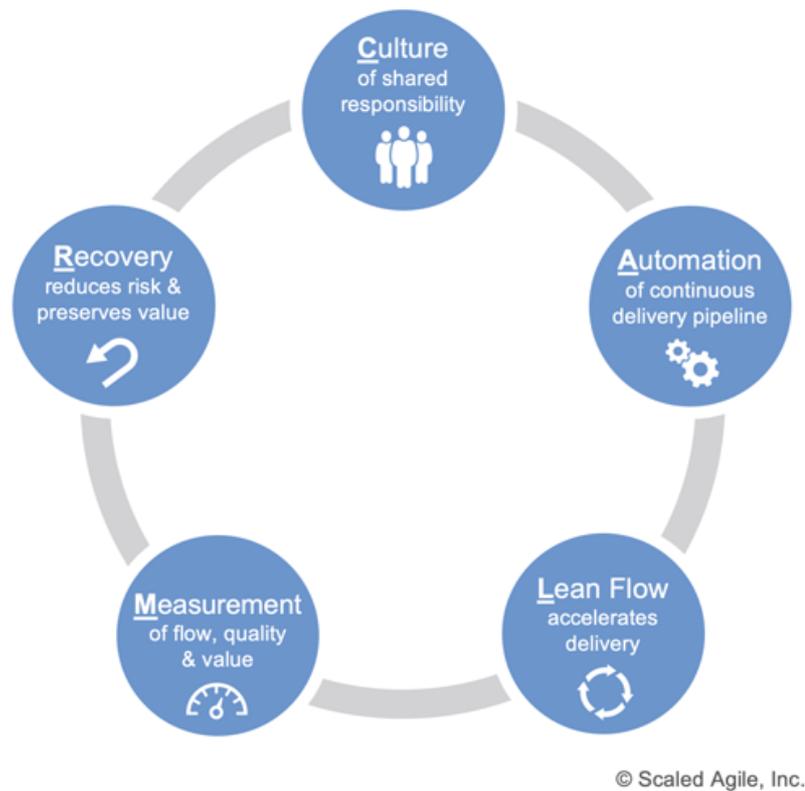


Figura 2.1.3 – Abordagem CALMR do *SAFe* para o DevOps

Fonte: [4]

**Cultura** – representa a filosofia de responsabilidade compartilhada para a entrega rápida de valor, com o apoio de todos que ajudam a criar valor como: gerenciamento de produto, desenvolvimento, teste, segurança, conformidade, operações.

**Automação** – necessidade de remover a intervenção humana ao máximo possível para diminuir erros e reduzir o tempo do ciclo.

**Fluxo Enxuto** – são as práticas de limitar o trabalho em processos (WIP), acelerando o fluxo de valor para o cliente e ter feedbacks mais rápidos.

**Medição** – estimula o aprendizado e melhoria contínua de fluxo, qualidade e valor.

**Recuperação** – permite soluções rápidas para resolver problemas de produção com reversão automática e recursos de correção antecipadas.

O CALMR foi projetado para trabalhar em organizações com qualquer nível de agilidade de negócios, apoiando imediatamente no início do processo.

### **3 – Team And Technical Agility (Equipe e Agilidade Técnica) [5]:**

Uma das competências essenciais de uma empresa enxuta. Equipe e Agilidade Técnica é essencial para Agilidade dos Negócios. Equipes ágeis apoiam as soluções que agregam valor ao negócio.

As equipes se unem em torno de um propósito compartilhado e aplicam modelo de entrega Lean-Agile baseado em fluxos. Os ARTs fornecem a estrutura necessária para alinhar as várias equipes em uma missão comum. A qualidade integrada garante que os membros da equipe tenham as habilidades necessárias para entregar valor, eliminando em grande parte, transferências e atrasos e práticas para criar as melhores soluções.

A equipe *SAFe* Agile é um grupo multifuncional de 5 a 11 pessoas que podem definir, construir, testar e fornecer um incremento de valor em um breve intervalo de tempo.

Os valores e princípios do manifesto ágil, embora originalmente concebidos para equipe de software, mostram que ter equipes de alto desempenho é importante para todos os tipos de negócio e tecnologias. Os insights listados abaixo podem ser aplicados e beneficiar qualquer equipe, são eles:

- Ser colaborativo;
- Enviar frequentemente;
- Usar medidas objetivas para o progresso;
- Interagir com o cliente com frequência;
- Esperar e apoiar mudanças.

As equipes ágeis podem empregar uma combinação de métodos ágeis, incluindo Scrum, XP e Kanban. Muitas configuram seus eventos de trabalho usando práticas Scrum, como:

- Trabalho em interações curtas, geralmente de duas semanas;
- Divide o trabalho em pequenas histórias de usuário gerenciadas pela equipe;
- Planeja o trabalho em conjunto para a próxima interação;
- Realizar eventos Daily Stand-up para comunicar e avaliar o andamento das metas da iteração;

- Demonstrar soluções de trabalho continuamente ou no final da iteração.

Para otimizar o fluxo, as equipes visualizam e gerenciam o andamento do trabalho utilizando quadros Kanban, como mostra o exemplo na figura 2.1.4. Esses quadros ajudam as equipes a identificar gargalos para melhorar o fluxo e limitar em processo WIP para garantir que as equipes terminem o trabalho antes de iniciar novas histórias.

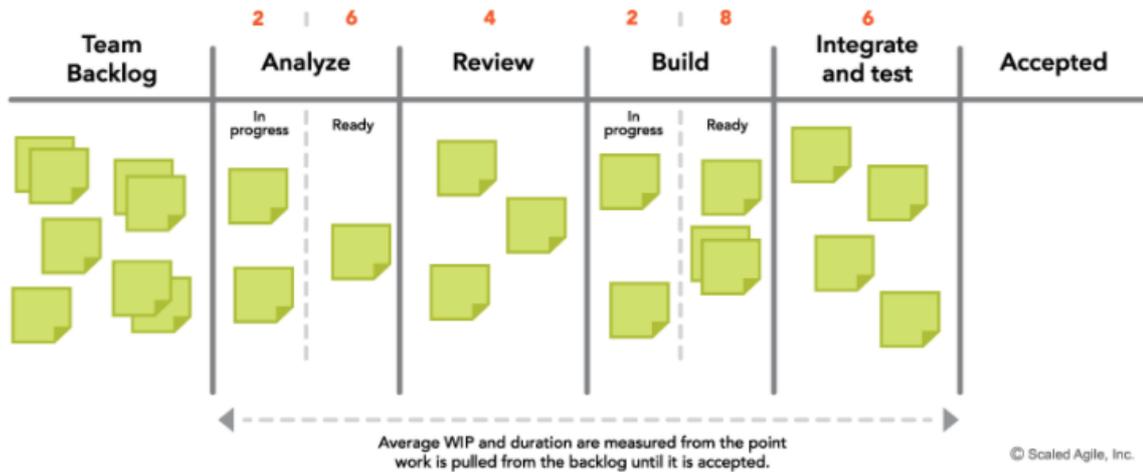


Figura 2.1.4 – Exemplo de um quadro Kanban

Fonte: [5]

Dependendo do tipo de negócio, uma equipe sozinha não tem capacidade de construir e entregar grandes sistemas dentro de um prazo razoável. Portanto várias equipes ágeis devem colaborar. O Agile Release Train (ART) do *SAFe* é uma longa equipe de equipes Agile, que junto com outras partes interessadas, desenvolve, entrega e, quando aplicável, opera uma ou mais soluções.

Todas as equipes Agile devem criar soluções de qualidade e definir suas próprias práticas para evitar retrabalho e atrasos.

#### 4 – Lean-Agile Leadership (Liderança Lean-Agile) [6]:

A competência Lean-Agile Leadership descreve como os líderes Lean-Agile conduzem a mudança organizacional e a excelência operacional ao capacitar indivíduos e equipes para alcançar seu potencial mais elevado. Os gerentes, executivos e demais líderes são responsáveis pela adoção, sucesso e melhoria contínua do desenvolvimento Lean-Agile e das competências que levam a agilidade ao negócio. Esses líderes podem criar um ambiente que incentive as equipes de alto desempenho, crescer e produzir valor, liderando pelo exemplo.

Tornar uma empresa Lean não é tarefa fácil, requer uma nova abordagem de liderança. Deve-se investir em treinamentos, capacitação e engajamento dos indivíduos e equipes para alcançar o objetivo por meio de princípios e práticas Lean e Agile.

O conhecimento por si só não é suficiente, os líderes Lean-Agile devem fazer mais que simplesmente apoiar a transformação, eles devem liderar ativamente a mudança, participando e orientando seus times nas atividades necessárias para compreensão e otimização contínua do fluxo de valor da empresa. Líderes Lean-Agile, devem:

- Organizar e reorganizar em torno do valor;
- Identificar filas e excessos de trabalho em processo (WIP);
- Concentrar na eliminação de desperdícios e atrasos;
- Eliminar políticas e procedimentos desmotivadores;
- Inspirar e motivar as pessoas;
- Criar uma cultura de melhoria implacável;
- Oferecer espaço para as equipes inovarem.

Ao ajudar os líderes a se desenvolverem, as organizações podem estabelecer a liderança ágil enxuta, como:

1 - Liderança pelo Exemplo: Líderes ganham autoridade ao modelar os comportamentos desejados, inspirando que outros líderes incorporem seu exemplo em seu próprio desenvolvimento pessoal.

2 - Mentalidade e Princípios: Inserir a maneira Lean-Agile de trabalhar em suas crenças, decisões, repostas e ações, ajudando a modelar o comportamento de toda organização.

3 - Liderando a Mudança: Liderar ao invés de simplesmente apoiar a transformação, criando o ambiente, preparando as pessoas e fornecendo os recursos necessários para atingir os resultados esperados.

O *SAFe* está alicerçado em quatro pilares: Lean, Agile, Pensamento Sistêmico e DevOps. O objetivo do *SAFe* foi desenvolver orientações para as empresas aplicarem os princípios e práticas do Lean e do Agile nas maiores organizações do mundo.

Lean é um conjunto de princípios e práticas para manufatura e operações eficientes que cresceram a partir do sistema Toyota de Produção, desenvolvido no Japão após a segunda guerra mundial. Ele se concentra na solução de problemas e na melhoria contínua para aumentar a qualidade e diminuir o desperdício.

Agile nasceu de uma colaboração de 17 líderes de pensamento em desenvolvimento de software que se reuniram em 2001 para buscar alternativas para os processos de

desenvolvimento de software pesados e orientados por documentação que eram comuns na época. O Agile inclui quatro valores [7] e dez princípios [8] [9].

### Valores:

1 - Alinhamento – O alinhamento é necessário para acompanhar as mudanças rápidas. Com equipes alinhadas é possível que todos tenham uma visão comum dos objetivos;

2 - Qualidade Embutida – Deve vir junto com o produto entregue. Ter a qualidade embutida é fundamental para garantir o sucesso da organização;

3 - Transparência – Agir com integridade e compartilhar informações de forma aberta, gerando confiança entre os membros das equipes e a organização como um todo. Os líderes Lean-Agile desempenham um papel fundamental na criação de um ambiente que estimula confiança e transparência;

4 - Execução do Programa – Entregar valor de forma contínua, como foco no resultado. Nada mais do *SAFe* importa se as equipes não podem executar e entregar valor continuamente.

### Princípios:

1 - Ter uma visão econômica – Esse princípio realiza o objetivo do Lean – Alcançar o menor tempo de espera com a melhor qualidade e valor. Hoje em dia não faz mais sentido gastar valores exorbitantes em projetos e além de tudo ter muita demora nas entregas.

Entregar mais cedo e com frequência é uma prática essencial para alcançar resultados satisfatórios com o uso do *SAFe*.

Escolher o caminho Lean-Agile é adotar um modelo baseado em desenvolvimento incremental e com entrega de valor contínua, como mostra o exemplo da figura 2.1.5.

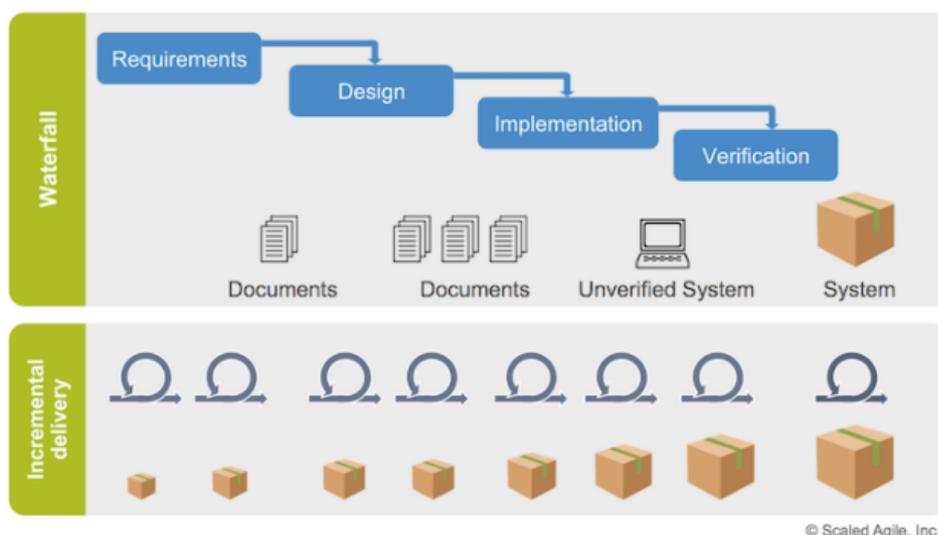


Figura 2.1.5 – Comparativo de mudança para entrega antecipada

Fonte: [10]

2 - Aplicar pensamento sistêmico – Os quatro aspectos básicos de conhecimento que compõe o *SAFe* são: Pensamento sistêmico, desenvolvimento ágil, desenvolvimento de produto enxuto e o DevOps.

3 - Assumir variabilidade; preservar opções – desenvolver sistemas e produtos de software é um trabalho incerto. A variabilidade técnica e a variabilidade do mercado estão presentes em todo processo de desenvolvimento. Para garantir o progresso, os desenvolvedores tendem reduzir a variabilidade, quanto mais determinadas são as coisas, mas seguros ficam os desenvolvedores. Gerenciar a variabilidade e preservar as opções, fornecendo controles e a flexibilidade de que as equipes precisam para construir grandes soluções.

4 - Construa de forma incremental com ciclos de aprendizagem rápidos e integrados – No desenvolvimento tradicional com controle de fases, os custos de investimento começam imediatamente e se acumulam até que uma solução seja entregue. Durante o desenvolvimento, é difícil obter feedbacks significativos, porque o processo não foi projetado para isso. Além disso, neste processo não foi implementado que o cliente avalie os incrementos com antecedência. Os pontos de integração precisam ser planejados em uma cadência regular para acelerar os ciclos rápidos de aprendizado. Quanto mais frequentes os pontos, mais rápido o aprendizado e quanto maior o sistema, mais esses níveis de integração existem, como ilustrado na figura 2.1.6.

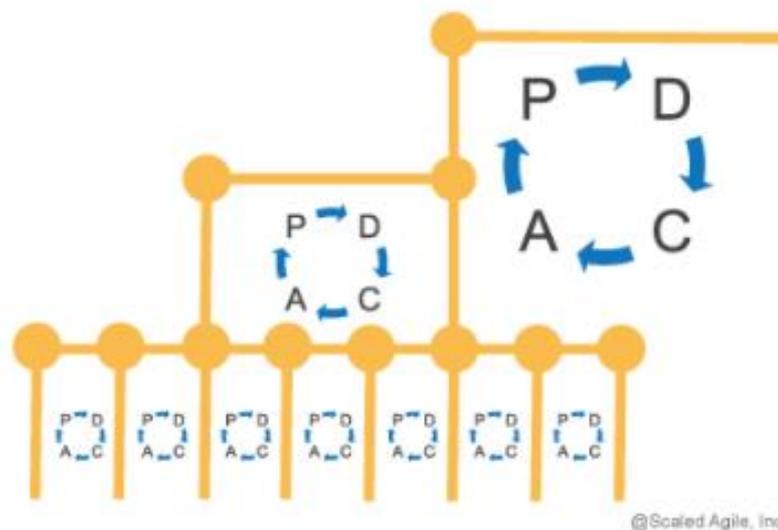


Figura 2.1.6 – Ciclos de PDCA (Plan, Do, Check, Act)

Fonte: [11]

5 - Marcos básicos na avaliação objetiva dos sistemas de trabalho – O desenvolvimento de sistemas atuais requer investimentos altos e os desenvolvedores e clientes têm a responsabilidade de garantir que os investimentos em novas soluções proporcionem os benefícios econômicos necessários. Geralmente é aplicado um processo de desenvolvimento sequencial com controle de fase (Waterfall) aparentemente lógico e sequencial: Identificação, requisitos, design, desenvolvimento, testes e implantação. Modelo este que nem sempre funciona muito bem. É necessário que seja feita uma abordagem diferente, como informa o Princípio 4. Construir sistemas com incrementos e validando gerando um incremento de valor.

6 - Visualize e limite o WIP, reduza os tamanhos dos lotes e gerencie os comprimentos das filas – Limitar o trabalho em processo ajuda as partes interessadas a ver bem como o trabalho está sendo executado. Sobrecarregar equipes com mais trabalho do que pode ser razoavelmente realizado é um problema comum. Ter muito WIP confunde as prioridades, causa frequentes trocas de contexto e aumenta a sobrecarga das pessoas, dispersa o foco em tarefas imediatas, reduz a produtividade e o rendimento. A primeira etapa para corrigir o problema é tornar o WIP visível para todas as partes interessadas. Utilizar um quadro Kanban é uma maneira simples de fazer isso, que ilustra a quantidade de trabalho em processo e o estado de cada item de trabalho, mostrando os gargalos atuais.

7 - Aplicar cadência, sincronizar com planejamento de domínio cruzado – Equipes ágeis naturalmente aplicam cadência nas Sprints ou iterações. Criar uma cadência para todos os assuntos possíveis reduz a complexidade, encaminha incertezas, reforça a qualidade e infunde colaboração. Sincronizar tais cadências permite que as pessoas e as atividades se movam como engrenagens na roda onde as informações aprendidas informam as decisões e o planejamento incremental.

8 - Desbloquear a motivação intrínseca dos trabalhadores do conhecimento – Importante para liberar o potencial das equipes. Muitas das vezes os colaboradores sabem mais sobre o trabalho que realizam que os próprios líderes. Com o *SAFe*, esses colaboradores são capazes de: Se comunicarem além dos limites funcionais; tomar decisões com base no entendimento econômico; receber feedback rápido sobre a eficácia da solução; participar de aprendizagem e domínio contínuo e incremental e participar de um processo de desenvolvimento de solução mais produtivo e gratificante.

9 - Descentralize a tomada de decisões – Agregar valor no prazo de entrega sustentável mais curto requer tomada de decisão descentralizadas. Algumas decisões são estratégicas, têm impacto de longo alcance e estão fora do escopo e responsabilidade das equipes, por isso os

líderes devem preservar sua autoridade para tomada de decisão para assuntos de importância estratégica e permitir que as equipes tomem escolhas a respeito dos demais aspectos.

10 - Organize em torno do valor – Focar no cliente é necessário e natural porque sem isso o fracasso nos negócios é rápido. O *SAFe* implementa um conjunto de fluxos de valores de desenvolvimento e fornece interfaces necessárias o equilíbrio do sistema, como: Reduzir transferências e atrasos entre as áreas funcionais, reduzindo o tempo de lançamentos no mercado; reunir todo o pessoal de pesquisa, desenvolvimento, implantação e serviço necessários para oferecer soluções completas de produtos; fornecer foco intenso no cliente; medir o sucesso por meio de indicadores e reorganizar rapidamente, conforme necessário, oferecendo suporte e oportunidades de ameaças competitivas.

## **5 – Continuous Learning Culture (Cultura de Aprendizagem Contínua) [12]:**

Cultura de Aprendizagem Contínua descreve um conjunto de valores e práticas que incentivam as pessoas e as empresas a aumentar continuamente o aprendizado, a competência, o desempenho e a inovação. As empresas hoje enfrentam muitas incertezas e oportunidades, com o ritmo acelerado de inovações tecnológicas. A qualquer momento, turbulências políticas, econômicas e ambientais ameaçam mudar as regras.

Toda organização pode iniciar a jornada de uma cultura de aprendizado contínuo focando em três dimensões críticas, como:

- Organização que aprende – Funcionários de todos os níveis aprendem e crescem para que a empresa possa se transformar e se adaptar num mundo de constantes mudanças;
- Cultura de Inovação – Funcionários são incentivados e capacitados a explorar e implementar ideias criativas que possibilitem a entrega de valor no futuro;
- Melhoria Implacável – Cada parte da empresa se concentra na melhoria contínua de suas soluções, produtos e processos.

O *SAFe* sugere algumas maneiras de promover aprendizado em uma organização, sendo alguma delas:

- Líderes Lean-Agile são aprendizes insaciáveis que usam tanto o sucesso quanto fracasso das práticas *SAFe* para momentos de aprendizado e desenvolvimento;
- Visão compartilhada é refinada interativamente durante cada período de planejamento;

- Equipes aprendem continuamente por meio da colaboração diária e da solução de problemas, apoiadas por eventos como retrospectivas da equipe e Inspect & Adapt;
- Pensamento sistêmico.

A base de uma cultura de inovação é o reconhecimento de que sistemas e cultura não inovam, o que inovam são as pessoas. Inculcar a inovação como uma capacidade organizacional central requer um compromisso em cultivar a coragem e a aptidão para encorajar a assunção de riscos entre os colaboradores. Isso pode exigir coaching, mentoria e treinamento formal nas habilidades e comportamentos de empreendedorismo e inovação.

## **6 – Organizational Agility (Agilidade Organizacional) [13]:**

Agilidade Organizacional descreve como pessoas com pensamento Lean e Equipes Agile otimizam seus processos de negócio e adaptam rapidamente a empresa conforme necessário para adquirir novas oportunidades. Na era atual é necessário ter velocidade para sentir e responder às necessidades dos seus clientes. Ter a capacidade de agregar valor no menor tempo de espera sustentável, evoluir e implementar novas estratégias.

O *SAFe* permite que as empresas se concentrem na inovação e no crescimento de novas ideias, utilizando três dimensões, que apoiam nesse processo de mudança.

- Pessoas com pensamento enxuto e equipes ágeis – todos os envolvidos na entrega de soluções são treinados nos métodos Lean e Agile e adotam seus valores, princípios e práticas;
- Operação de Negócios Enxuta – as equipes aplicam princípios Lean para compreender, mapear e melhorar continuamente os processos que fornecem e oferecem suporte às soluções de negócios;
- Agilidade da Estratégia – A empresa é ágil o suficiente para sentir continuamente o mercado e mudar rapidamente a estratégia quando for necessário.

Entender a mentalidade Lean Agile é a base de uma nova abordagem de gerenciamento e resulta em uma cultura corporativa aprimorada que permite agilidade nos negócios. Os dez princípios básicos do *SAFe* que foram descritos mais acima orientam funções, práticas e comportamentos eficazes. Todos na empresa podem aplicar esses princípios em seu dia a dia e assim se tornar parte de um sistema operacional enxuto e ágil.

Sem agilidade organizacional, as empresas não podem reagir rapidamente quando as coisas acontecem. As operações de negócios enxutas reconhecem que satisfazer os clientes vai

além do desenvolvimento de soluções, toda jornada do cliente precisa ser continuamente otimizada para reduzir o tempo de lançamentos no mercado e aumentar a satisfação do cliente.

## **7 – Lean Portfolio Management (Gestão Enxuta de Portfólio) [14]:**

Alinha estratégia e execução aplicando abordagens Lean e de pensamentos sistêmico à estratégia e financiamento de investimentos, operações de portfólio Agile e Governança, como descrito a seguir:

**Estratégia e Financiamento de Investimento** garante que todo portfólio esteja alinhado e financiado para criar e manter as soluções necessárias para cumprir as metas do negócio.

**Operação de Portfólio Ágil** coordena e apoia a execução descentralizada do programa e promove a excelência operacional. Os princípios do *SAFe* e a mentalidade Lean-Agile promovem a descentralização da execução da estratégia para os Agile Release Train (ARTs), portanto o pensamento sistêmico deve ser aplicado para garantir que os ARTs estejam alinhados e operem dentro do contexto empresarial mais amplo.

**Governança Enxuta** é a supervisão e tomada de decisão de gastos, auditoria e conformidade, previsão de despesas e medição. A colaboração e as responsabilidades da governança Lean exigem o envolvimento ativo do Agile PMO, proprietários de negócio e arquitetos corporativos.

Definir e executar com sucesso uma estratégia em um mundo de crescentes incertezas é um grande desafio. Requer práticas de gerenciamento de portfólio modernizadas pela aplicação do pensamento Lean-Agile e um portfólio organizado em torno de fluxos de valor que fornecem um fluxo contínuo de valor para os clientes da empresa.

As sete competências apresentadas ao longo deste capítulo trabalham juntas para criar resultados econômicos e gerar entrega de valor para a companhia e para seus clientes.

## **2.2 – Trabalhando com o Kanban**

Kanban é um termo de origem japonesa e significa “cartão” ou “sinalização” [15].

“O Kanban usa um mecanismo de controle visual para acompanhar o trabalho à medida que ele flui através das várias etapas do fluxo de valor” (ANDERSON; DAVID J, 2009, p. 16, grifo do autor). [16]

O método Kanban não é apenas um quadro de tarefas com post-it coloridos.

O Kanban usa o conceito de cadeia puxada como ferramenta principal para expor problemas, gerar questionamentos e criar melhorias no fluxo de trabalho.

O quadro Kanban ficou muito conhecido após a chegada do framework SCRUM, onde é usado para representar o andamento das atividades auxiliando na gestão visual das atividades, como mostra o exemplo da figura 2.2.1.



Figura 2.2.1 – Quadro Kanban

Fonte: [17]

Estimula desenvolvimento e transformação gradual e progressiva, além de crescimento e aperfeiçoamento de processos e sistemas. Uma forma simples e econômica de controlar a capacidade de trabalho dos times. O uso do quando Kanban tem vantagens como:

- Definição do fluxo de trabalho;
- Organização dos processos de acordo com o tema;
- Definição dos responsáveis de cada demanda;
- Visualização da complexidade do processo assim como tempo gasto para conclusão;
- Mapeamento das dependências.

### 2.2.1 Princípios do Kanban

Em 2016, David Anderson ampliou e separou os princípios do método Kanban em dois grupos:

#### **Princípios da Gestão da Mudança [18]**

1. Comece com o que você já faz hoje: O método Kanban não prega revolução e sim evolução, então mapeie o seu fluxo de valor atual. Não existe certo ou errado, tudo

depende do contexto. Seu contexto dirá se o processo funciona em sua organização ou não.

2. Busque mudanças evolucionárias e incrementais: Após trabalhar um período com seu fluxo de valor mapeado, busque pequenas mudanças. A prática do Kaisen pode auxiliar, pois incentiva a busca da melhoria contínua, que é um processo sem fim, no qual sempre estamos melhorando de alguma forma.
3. Incentive atos de liderança em todos os níveis: Algumas das melhores lideranças vêm de atos cotidianos de pessoas na linha de frente de seus times. É importante que todos promovam uma mentalidade de melhoria contínua, a fim de atingir o desempenho ideal em um nível de time/departamento/organização.

### **Princípios da Entrega de Serviços**

4. Foque no cliente: Entenda e foque nas expectativas e necessidades do cliente. Faça questionamentos como: Por que queremos melhorar? O que estamos tentando melhorar? É preciso entender claramente quem é o cliente, o que ele solicita e quais são as suas expectativas em termos de serviço.
5. Gerencie o trabalho e deixe as pessoas se auto organizarem: Não gerencie pessoas ou o tempo que essas pessoas possuem para realizar o trabalho, gerencie o trabalho, seja tangível ou intangível (trabalho do conhecimento).
6. Desenvolva e evolua políticas para melhorar os resultados de clientes e negócios: Esse princípio está diretamente ligado a terceira prática do método Kanban (Torne as políticas do processo explícitas), ao invés de visualizar os processos como uma sequência de atividades, fluxo de trabalho, funções e responsabilidades, visualize seus processos existentes como conjuntos de políticas e essas políticas devem ser definidas explicitamente.

### **2.2.2 Benefícios do Kanban**

Sistema Puxado – A equipe tem autonomia para assumir um trabalho pois o backlog sempre está priorizado;

Fluxo contínuo – Não possui ciclos de entrega;

Transparência – Todos podem enxergar o contexto do outro, levando instantaneamente ao aumento da comunicação e da colaboração;

Previsibilidade – Visão ampla do que está sendo feito, em qual etapa, o que está pronto, quanto está pronto e o quanto a equipe consegue entregar.

### **2.2.3. As 6 práticas do método Kanban**

1 - Visualize o fluxo de trabalho: Fracione o fluxo de trabalho em partes, escreva cada atividade em um cartão/post-it e tenha visão de todo o trabalho que está em andamento.

2 - Limite o trabalho em progresso (WIP): Ao limitar o WIP, o ritmo da equipe se torna equilibrado, porque os profissionais não são comprometidos com muito trabalho de uma só vez, é reduzido o tempo gasto em um item e evita problemas com alternância de tarefas, assim mantendo o Lead Time estável.

3 - Torne as políticas do processo explícitas: Não é possível melhorar algo que você não entende. Ao trabalhar com o método Kanban, você vai perceber que muitas regras de processos, papéis e definições não são claras para os envolvidos, então crie, divulgue e socialize o processo, assim todos terão uma clareza de como ele funciona e de como o trabalho realmente é feito.

4 - Meça e gerencie o fluxo: Após ter a visibilidade do fluxo de trabalho, determinar o limite WIP, conhecer e deixar as políticas do processo de forma explícita, será preciso ter visibilidade de quanto tempo as atividades que passam pelo fluxo de trabalho têm levado para serem concluídas, além de remover os possíveis impedimentos que por acaso estejam bloqueando o fluxo.

5 - Implemente mecanismos de feedback: Ajuste as reuniões existentes, bem como a introdução de novas reuniões para acomodar os ciclos de feedback, conhecidas como as Cadências do Kanban. Reduza o tempo do ciclo de feedback, pois quanto mais tardia a entrega, mais demorada será a geração de valor daquilo que foi trabalhado.

6 - Melhore colaborativamente, evolua experimentalmente: Visto que agora temos a visibilidade de onde estão os nossos problemas e gargalos, é preciso melhorar o nosso fluxo de trabalho. Torne os problemas visíveis e engaje o grupo emocionalmente na mudança.

### **2.2.4. Cadeia Puxada com Gestão Visual**

Gestão visual é uma ferramenta importante para identificar gargalos nos processos e projetos. Utilizar técnicas de gestão visual aumenta o nível de engajamento dos colaboradores,

tornando possível que a equipe se torne autônoma e auto organizada, sem depender que o gestor precise delegar as atividades do dia a dia.

Premissas para utilização de Cadeira Puxada:

**Transparência** – Necessidade de visibilidade das demandas de processos recebidas, a fim de acompanhar e reportar as entregas de valor priorizadas;

**Priorização** - Necessidade de ter visibilidade da prioridade das demandas de processos;

**Interdependência** - Necessidade de identificação e mapeamento da dependência entre as demandas de processos;

**Capacidade** - Necessidade de visibilidade do backlog e conhecimento da capacidade do time de (WIP) de acordo com a volumetria e complexidade das demandas de processos.

## 2.3 – Características do Scrum

De acordo com o Scrum Guide [19], desenvolvido e mantido por Ken Schwaber e Jeff Sutherland, Scrum é um framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos. Scrum não é um processo ou técnica para construir produtos, sim um framework dentro do qual pode ser empregado vários processos e técnicas.

O framework Scrum consiste em times associados a papéis, eventos, artefatos e regras.

Scrum é fundamentado nas teorias empíricas de controle de processo, onde o conhecimento vem da experiência e de tomada de decisões baseadas no conhecimento. O Scrum emprega uma abordagem interativa e incremental para aperfeiçoar a previsibilidade e o controle de riscos, sendo apoiado por três pilares: Transparência, Inspeção e Adaptação.

**Transparência:** Aspectos significativos do processo devem estar visíveis aos responsáveis pelos resultados;

**Inspeção:** Usuários Scrum devem, frequentemente, inspecionar os artefatos e o progresso com objetivo de detectar variações, sem que essas inspeções atrapalhe a execução das tarefas;

**Adaptação:** Se durante a inspeção for identificado que um processo desviou fora dos limites aceitáveis ou o resultado do produto está inaceitável, o processo ou produto que está sendo produzido deve ser ajustado o mais breve possível para minimizar mais desvios.

O Scrum indica eventos formais, conforme demonstra a figura 2.3.1, para criar uma rotina e minimizar a necessidade de reuniões não definidas. São eventos com tempo definido, de tal modo que todos tenham um tempo máximo para acontecer. Esses eventos são contidos

dentro dos limites da Sprint. A Sprint é o coração do Scrum, tem duração definida para atender o esforço do desenvolvimento. Uma Sprint começa quando a anterior finaliza.

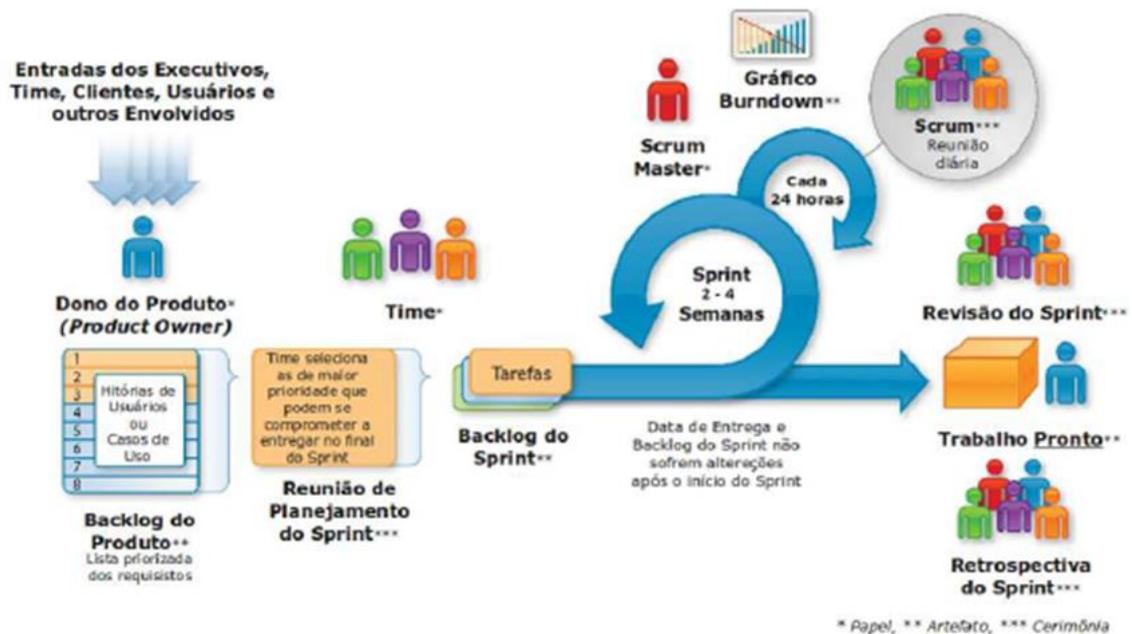


Figura 2.3.1 – Fluxo Scrum

Fonte: [20]

Cada evento no Scrum é uma oportunidade de inspecionar e adaptar alguma coisa, são eventos especificamente projetados para permitir transparência e inspeção criteriosa.

Eventos sugeridos pelo Scrum:

**Reunião de planejamento da Sprint** – O trabalho a ser realizado numa Sprint é definido na reunião de Planejamento da Sprint. Essa reunião tem um tempo de duração de no máximo 8 horas para Sprint de um mês de duração, para Sprints menores, este evento costuma ser menor. Durante a reunião são definidas questões como: O que pode ser entregue como incremento da próxima Sprint? E, como o trabalho necessário para entregar o incremento será realizado?

**Reunião diária** – É um evento diário com duração de quinze minutos para que o time possa sincronizar as atividades e criar um plano para as próximas vinte e quatro horas. É uma reunião que deve ser mantida sempre no mesmo horário e local. Durante a reunião os membros dos times informam:

- O que foi feito ontem que ajudou o time a atender a meta da Sprint;
- O que será feito hoje para atender a meta da Sprint;
- Quais os obstáculos que podem impedir o atendimento da meta da Sprint.

**Reunião de revisão da Sprint** – Realizada no final da Sprint para inspecionar o incremento e adaptar o backlog do produto, se for necessário. Durante a reunião o time Scrum e partes interessadas colaboram sobre o que foi feito na Sprint. É uma reunião com tempo definido de quatro horas de duração para uma Sprint de um mês. A reunião de revisão inclui itens como:

- Participação do time Scrum, Stakeholders-chaves e Product Owner (PO);
- PO esclarece quais itens do backlog estão “prontos” e quais não foram;
- Time de desenvolvimento coloca o que foi bem durante a Sprint, quais os problemas ocorreram e como os mesmos foram resolvidos;
- Time de desenvolvimento demonstra o trabalho que está “pronto”;
- Todo time colabora sobre o que fazer a seguir e com isso gera insumos para a próxima reunião de planejamento da próxima Sprint.

**Retrospectiva das Sprint** – Este evento é importante pois gera oportunidade para Time Scrum inspecionar a si próprio e criar um plano para melhorias para a próxima Sprint. Ocorre depois da Revisão da Sprint e antes da reunião de Planejamento da próxima Sprint. O propósito da Retrospectiva é:

- Inspeccionar como a última Sprint foi em relação às pessoas, aos relacionamentos, aos processos e ferramentas;
- Identificar e ordenar os principais itens que foram bem e as potenciais melhorias;
- Criar um plano para implementar melhorias no modo que o time fez o seu trabalho.

A figura 2.3.2, mostra o exemplo de uma dinâmica que pode ser utilizada durante a Retrospectiva da Sprint, onde os membros do time colocam suas percepções em relação ao trabalho realizado. Com essa ilustração pode ser analisado o resultado da Sprint e o que se passou durante a jornada, como: O que impulsionou a jornada (ventos), o que gerou impedimentos (âncora), risco que não estavam previstos (iceberg), o que o time apreciou durante a Sprint, que funcionou bem e que pode ser repetido nas demais (Sol), qual foi a percepção do time com o resultado alcançado (Ilha).

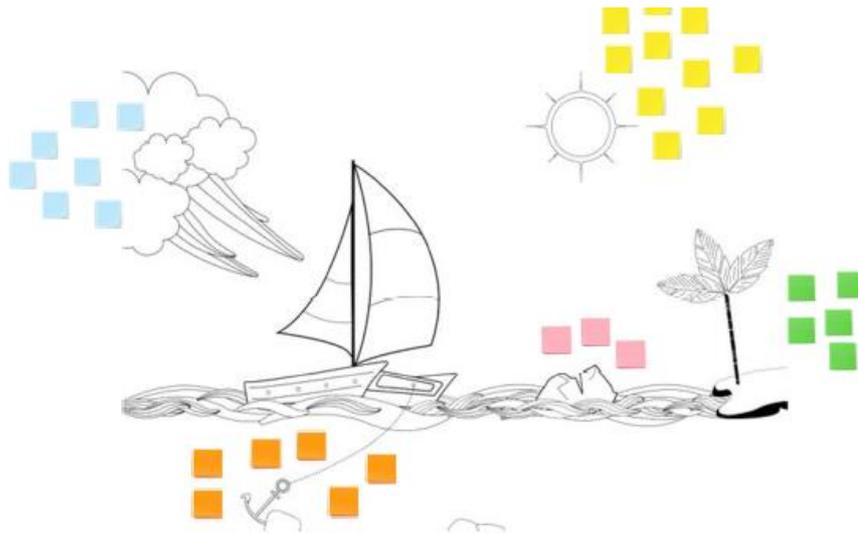


Figura 2.3.2 – Exemplo da Dinâmica que pode ser feita durante a Retrospectiva da Sprint

Fonte: [20]

No Scrum os artefatos representam o trabalho e o valor para o fornecimento de transparência e oportunidades para inspeção e adaptação. Eles são projetados para que as informações sejam transparentes e permita que todos tenham o mesmo entendimento. A seguir alguns artefatos que são produzidos:

**Backlog do Produto** – É uma lista ordenada de tudo que deve ser necessário no produto, e é uma origem única dos requisitos de qualquer mudança a ser feita no produto. O PO é o responsável pelo Backlog do Produto, incluindo o seu conteúdo, disponibilidade e ordenação. Backlog do Produto é dinâmico, pode mudar constantemente para identificar o que o produto necessita para ser mais apropriado, competitivo e útil.

**Backlog da Sprint** – É um conjunto de itens do Backlog do Produto 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 que será necessário realizar para entregar essa funcionalidade em um incremento “pronto”. O Backlog da Sprint torna visível todo o trabalho que o time de desenvolvimento identifica como necessário para atingir o objetivo da Sprint, é um plano com detalhes suficientes para que as mudanças no progresso sejam entendidas durante a Reunião Diária.

**Monitorando o Progresso da Sprint** – O time de desenvolvimento monitora o total do trabalho restante pelo menos a cada Reunião Diária, acompanhando estes resumos diários e projeto a probabilidade de alcançar o objetivo da Sprint;

**Incremento** – É a soma de todos os itens do Backlog do Produto completados durante a Sprint e o valor dos incrementos de todas as Sprints anteriores. Ao final da Sprint um novo incremento deve estar “Pronto”, o que significa que deve estar na condição utilizável e atender a definição de “Pronto” do Time Scrum.

O time Scrum é composto pelo Product Owner, Time de Desenvolvimento e Scrum Master. Times Scrum são auto organizáveis e multifuncionais. Times auto organizáveis escolhem qual a melhor forma para completarem seu trabalho. Times multidisciplinares possuem competências necessárias para completar o trabalho sem depender de outros que não fazem parte do time.

Abaixo serão descritas algumas características dos papéis no Scrum:

**Product Owner (PO)** – É o dono do produto, responsável por gerenciar o Backlog do Produto, o que inclui:

- Expressar claramente os itens do Backlog do Produto;
- Ordenar os Itens do Backlog;
- Garantir que o Backlog do Produto seja visível, transparente, claro para todos e mostrar o que o time fará a seguir;
- Garantir que o time de desenvolvimento entenda os itens do Backlog do Produto.

**Time de Desenvolvimento (Dev Team)** – Consiste de profissionais que realizam o trabalho de entregar uma versão usável que potencialmente incrementa o produto “Pronto” ao final de cada Sprint. O Time de Desenvolvimento tem as seguintes características:

- Eles são auto organizados;
- Times de Desenvolvimento são multifuncionais, possuindo todas as habilidades necessárias, enquanto equipe, para criar o incremento do Produto;
- O Scrum não reconhece títulos para os integrantes do Time de Desenvolvimento que não seja o Desenvolvedor, independentemente do trabalho que está sendo realizado pela pessoa;
- Individualmente os integrantes do Time de Desenvolvimento podem ter habilidades especializadas e área de especialização, mas a responsabilidade pertence ao Time de Desenvolvimento como um todo;
- Times de Desenvolvimento não contém sub times dedicados a domínios específicos de conhecimento, tais como teste ou análise de negócios.

**Scrum Master (SM)** – Responsável por garantir que o Scrum seja entendido e aplicado. É um servo líder para o time Scrum, trabalha com os demais papéis de várias maneiras, como:

- Atuação junto ao PO – O SM trabalha junto com o PO encontrando técnicas para o gerenciamento efetivo do Backlog do Produto; comunica de forma clara o objetivo os itens do backlog para o Time de Desenvolvimento; explica como o Time Scrum deve criar itens no Backlog do Produto, de forma clara e concisa; apoia na compreensão e prática da agilidade; facilita os eventos Scrum conforme exigidos e necessários;
- Atuação junto ao Time de Desenvolvimento – Treina o Time de Desenvolvimento em autogerenciamento; ensina e lidera o Time de Desenvolvimento na criação de produtos de alto valor; remove impedimentos para o progresso do Time de Desenvolvimento; facilita os eventos Scrum conforme exigido ou necessário; treina o Time de Desenvolvimento em ambientes organizacionais nos quais o Scrum não é totalmente adotado e compreendido;
- Atuação junto a Organização – Lidera as iniciativas de mudanças, treinando a organização na adoção do Scrum; planeja implementações Scrum dentro da organização; ajuda funcionários e partes interessadas a compreender e tornar o Scrum aplicável.

O Scrum é livre, leve e adaptável. É uma forma diferente de trabalhar que completada com outras técnicas funciona muito bem.

## **2.4 – XP – Extreme Programming**

O XP [21] [22], é considerado uma metodologia ágil que se ajusta bem em projetos de software que tenham requisitos vagos e em constante mudança. Adota a estratégia de constante acompanhamento e realização de pequenos ajustes durante o desenvolvimento.

Tem características marcantes como:

- Feedbacks constantes;
- Abordagem incremental;
- Encoraja a comunicação entre as pessoas envolvidas.

E tem cinco valores fundamentais como:

- Comunicação;
- Simplicidade;
- Feedback;
- Coragem;

- Respeito.

E a partir desses valores possui princípios básicos, como:

- Feedback rápido;
- Presumir simplicidade;
- Mudanças incrementais;
- Aceitar as mudanças;
- Trabalho de qualidade.

Dentre as variáveis de controle de projetos (custo, tempo, qualidade e escopo), o XP tem um foco explícito em Escopo, por isso é recomendada a priorização de funcionalidades que representem maior valor para o negócio. O XP incentiva o controle de qualidade como variável do projeto, pois o pequeno ganho de curto prazo da produtividade, ao diminuir a qualidade, não é compensado por perdas ou até mesmo impedimentos ao longo prazo.

Para aplicar valores e princípios durante o desenvolvimento o XP propõe uma série de práticas, como as exemplificadas abaixo.

**Jogo de Planejamento** – O desenvolvimento do produto é feito em interações semanais. Antes de iniciar o desenvolvimento, o time de desenvolvimento e usuários se reúnem para priorizar as funcionalidades através das User Stories (Estórias de Usuário). Nesta reunião são identificadas as prioridades e estimado o tempo. Os usuários são essenciais neste processo, pois é através dele que eles ficam sabendo o que está acontecendo e o que vai acontecer no projeto;

**Pequenas Entregas** – A liberação de pequenas versões funcionais do projeto auxilia muito no processo de aceitação dos clientes, que já podem testar uma parte do produto que está comprando;

**Teste** – A prática de testes no XP é bastante técnica e envolve a presença de clientes no desenvolvimento e na validação dos testes. O cliente compartilha com o desenvolvedor sobre o funcionamento do sistema. Os testes também se tornam as especificações do desenvolvedor, uma vez que os testes são escritos antes da funcionalidade, também conhecido como TDD (Test Driven Design), onde intercala a função de testar um pouco e codificar um pouco;

**Programação em Pares** – Se trata de duas pessoas trabalhando numa máquina onde um codifica e o outro faz as críticas ou dá sugestões. Os pares trocam de lugar periodicamente favorecendo comunicação e aprendizado. Essa prática resulta em projetos com maior qualidade, maior produtividade;

**Semana de 40 horas** – Trabalhar com qualidade, buscando um ritmo de trabalho saudável de 40 horas por semana – 8 horas/dia, sem horas extras. Horas extras são permitidas quando trouxerem produtividade para a execução do projeto.

A transição para o XP não é simples e exige habilidade o que pode ser um desafio para muitos. A comunicação tem que ser muito forte e exige uma mudança cultural importante para que sua implementação dê certo, mas não é impossível, basta que todos os envolvidos queiram mudar.

## 2.5 – Conceitos de Teste de Software

Teste de Software [23] é uma importante etapa do ciclo de desenvolvimento de um produto, pois é feito um controle de qualidade que pode envolver validações e verificações em várias etapas do processo. Habitualmente o teste de software é realizado no final da cadeia, onde são realizados os testes para validação do produto desenvolvido antes da disponibilização do mesmo em produção. Portanto quanto mais cedo começar a validação e verificação do produto, serviço ou sistema, mais cedo são identificados erros que podem comprometer a qualidade do que está sendo entregue, isso interfere também no custo e no risco de problemas em produção. Quanto mais tarde identificar um erro, maior o custo para o projeto.

Teste é um processo que permeia outros processos da engenharia de software e envolve ações que vão desde o levantamento de requisitos até a execução dos testes em si.

Não se pode garantir que um software funcione corretamente, sem a presença de erros.

A norma ISO/IEC 9126 [24] para qualidade de produto de software define atributos qualitativos como:

**Funcionalidade** – Capacidade de um software prover funcionalidade que satisfaçam o usuário em suas necessidades declaradas e implícitas, dentro de um determinado contexto de uso, como:

- Adequação;
- Acurácia;
- Interoperabilidade;
- Segurança;
- Conformidade.

**Confiabilidade** – Capacidade de um software manter um nível de desempenho adequado, quando usado em situações especificadas;

**Usabilidade** – É a capacidade de um produto de software ser compreendido, operado e atraente ao usuário. Tem características específicas, como:

- Inteligibilidade;
- Operacionalidade;
- Estética;
- Acessibilidade;
- Conformidade.

**Eficiência** – Quando o tempo de execução e os recursos envolvidos são compatíveis com o nível de desempenho do software. Tem características, como:

- Comportamento em relação ao tempo de resposta em relação ao que foi especificado;
- Utilização de recursos que mede tanto os recursos consumidos quanto a capacidade do sistema em utilizar os recursos disponíveis de processador e memória, por exemplo;
- Conformidade, que é a capacidade de um produto estar de acordo com normas, convenções ou regulamentações relacionadas à eficácia.

**Manutenibilidade** – Facilidade de um produto de software ser modificado, incluindo tanto as melhorias ou extensões de funcionalidades quanto as correções de defeitos;

**Portabilidade** – Capacidade de um sistema ser transferido de um ambiente para outro. Tem algumas características, como:

- Adaptabilidade – Capacidade do software se adaptar a diferentes ambientes sem a necessidade de ações adicionais;
- Capacidade de Instalação – Facilidade com que o software pode ser instalado em um novo ambiente;
- Coexistência – Mede quão facilmente um software convive com outros instalados no mesmo ambiente.

**Qualidade em uso** – Capacidade de um produto atingir metas específicas de eficácia, produtividade, segurança e satisfação, seguindo algumas características como:

- Eficácia – Quando o produto de software permite que os usuários atinjam metas especificadas com acurácia e completude;
- Segurança – Capacidade do produto de software de apresentar níveis aceitáveis de riscos de danos a pessoas, negócio, software ou ambiente;
- Satisfação – Quando o produto de software satisfaz a solicitação do usuário.

Mensurar o bom funcionamento de um software envolve compará-lo com elementos como especificações, outros softwares da mesma linha, versões anteriores do mesmo produto, expectativas do cliente, normas e leis aplicáveis. Enquanto a especificação diz respeito ao processo de verificação do software, a expectativa do cliente diz respeito ao processo de validação do software.

Por meio da verificação será analisado se o produto foi feito corretamente, se ele está de acordo com os requisitos especificados. Por meio da validação será analisado se foi feito o produto correto, se ele está de acordo com as necessidades e expectativas do cliente.

Existem princípios vitais para o Teste de Software como por exemplo os Casos de Testes, que define critérios de entrada, processamento e saídas esperadas. Os casos de testes devem considerar não somente condições válidas, como condições inválidas também, com testes positivos e negativos.

São várias as maneiras de testar um software, abaixo serão descritas algumas técnicas utilizadas:

- Teste de Caixa-Branca – Também conhecido como teste estrutural ou orientado a lógica, este tipo de teste avalia o comportamento interno do software;
- Teste de Caixa-Preta – São os testes funcionais, comportamental, orientado a dados ou com critérios de entrada e saída;
- Teste de Regressão – Técnica aplicável em uma nova versão do software ou quando é necessário executar um novo ciclo de teste durante o processo de desenvolvimento, consiste em aplicar todos os testes que já foram realizados em versões ou ciclos anteriores;
- Testes Não Funcionais – Utilizado para verificação do comportamento do sistema em relação a casos inválidos ou inesperados de entrada. Nessa técnica é muito utilizado os testes de carga e testes de desempenho, que verifica se o software consegue processar grandes quantidades de dados e no tempo de processamento especificados.

Uma prática comum é testar o software após uma funcionalidade ser desenvolvida e antes dela ser implementada. Os testes podem ser executados em fases diferentes e com objetivos diferentes, mas que ao final resulta na garantia da qualidade do produto que está sendo entregue. Abaixo algumas das fases de testes:

- Teste Unitário – Fase em que é testada a menor unidade do código desenvolvido, tem objetivo de encontrar falhas de funcionamento dentro de uma pequena parte do sistema independente do todo;
- Teste de Integração – Objetivo encontrar falhas provenientes da integração interna dos componentes de um sistema;
- Teste de Sistemas – Executar testes sobre o ponto de visto do usuário final, varrendo as funcionalidades em busca de falhas em relação aos objetivos do software. São testes realizados em condições similares de ambiente, interfaces sistêmicas e massa de dados, que os usuários irão utilizar no seu dia-dia;
- Testes de Aceitação – Geralmente são realizados por um grupo de usuários finais do sistema, que simula operações de rotina do sistema de modo a verificar se o software está se comportando conforme solicitado;

Importante que as empresas tenham um processo de testes e que este processo tenha definido as fases para preparação e execução dos testes, o ciclo de vida dos testes é composto pelas seguintes fases:

- Planejamento – Etapa de elaboração da Estratégia de Testes e o Plano de Testes de acordo com os requisitos especificados;
- Preparação – Fase onde é preparado o ambiente para a execução dos testes, como: equipamentos, pessoal, ferramentas de automação, massa de testes;
- Especificação – Nessa fase é elaborado e/ou revisado os casos de testes;
- Execução – Fase em que os testes são executados de acordo com os casos de testes e a registrado os resultados obtidos; nesta etapa, os defeitos são identificados, corrigidos e retestados até que o resultado esperado esteja de acordo com o desejado;
- Entrega – Última fase do ciclo de vida dos testes, onde o projeto é finalizado e toda documentação referente aos testes é finalizada e arquivada.

Testes de Software é uma etapa importantíssima no Ciclo de Vida de Desenvolvimento de Software, no entanto ainda é uma disciplina pouco reconhecida pela sua importância, muitas vezes entendida como uma etapa que atrapalha o processo de desenvolvimento. Mudar esse paradigma é de suma importância para que as empresas tenham cada vez mais qualidade nas suas entregas.

## Capítulo 3

# Estudo de Caso – Implantando a Transformação Ágil na Diretoria de TI

Este capítulo irá abordar o Estudo de Caso que será tratado neste trabalho. Neste capítulo será apresentado como o processo de transformação ágil está inicialmente planejado na diretoria de TI de uma empresa de telecom. Este trabalho terá como foco principal apresentar impacto dessa transformação numa área específica da TI, que é a área de Gestão de Testes, que faz parte da Gerência de Qualidade, conforme exemplificado na figura 3.1.

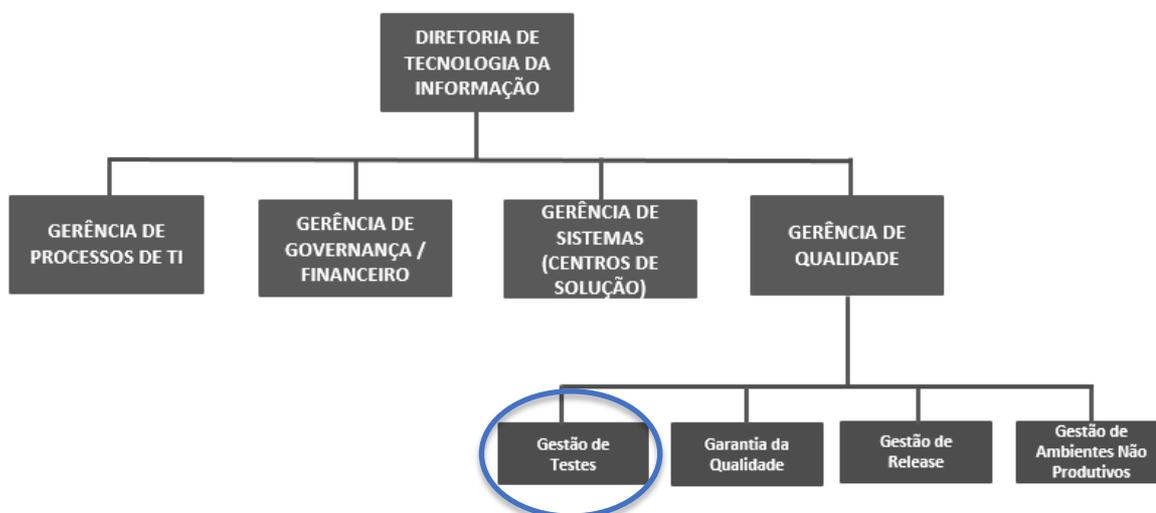


Figura 3.1 – Estrutura Gerencial da Diretoria de TI da Empresa de Telecom

Fonte: Autor

Até o final de 2019 a diretoria de TI atuava nos seus projetos com o modelo Waterfall para desenvolvimento de software. Todas as atividades eram feitas de forma sequencial e quando uma etapa do projeto avançava, áreas que já haviam executado suas atividades não continuavam participando ativamente do projeto até o seu término. Faltava engajamento do time como um todo para o real sucesso dos projetos, pois cada um fazia sua parte, mas nenhuma área se comprometia com o todo. A demora na entrega dos projetos era muita grande e por várias vezes quando um determinado projeto era entregue já não tinha mais nenhum valor para

a companhia. A área de testes era envolvida tardiamente nos projetos, muitas das vezes só lembravam de impactar a área quando o projeto já estava para iniciar a execução dos testes. Diante disso com a concorrência acirrada e a necessidade de estar à frente nos seus negócios, a empresa sentiu a necessidade de rever seus processos e se modernizar.

Com isso no início de 2020 foi dada a largada para a nova reestruturação dos processos da diretoria de TI para implantação do Ágil.

Abaixo será descrito como o processo ágil foi desenhado e na sequência será apresentado o processo da área de testes dentro deste contexto.

### 3.1 Transformação Ágil – Implantação do *SAFe*

Durante a reestruturação do processo foi definido que o modelo de governança irá funcionar com três camadas de gestão, sendo elas:

- Originação;
- Gestão Técnica;
- Desenvolvimento.

**Originação** - Etapa onde é definido o *Backlog* da Iniciativa, nessa etapa as seguintes atividades são realizadas:

- Alinhamento com a estratégia de Negócio;
- Elaboração das Features (funcionais e técnicas) das iniciativas candidatas;
- Elaboração do desenho da arquitetura de solução;
- Priorização iniciativas e alocação nos trens;
- Definição do MVP;
- Priorização das Features para a próxima Big Room Planning (BRP).

**Gestão Técnica** - Camada de acompanhamento técnico, onde são realizados os refinamentos funcionais e técnicos das *Features* priorizadas e onde a equipe de testes inicia sua participação com a escrita da Estratégia de Testes. As atividades que compõem essa camada são:

- Refinamento funcional e técnico das Features priorizadas;
- Escrita das histórias das Features para o próximo ciclo (BRP);
- Refinamento das histórias para o próximo ciclo (BRP);
- Elaboração da Estratégia de Teste de Projeto.

**Desenvolvimento** - Camada de refinamento das histórias, com as seguintes atividades:

- Refinamento final das estórias;
- Desenvolvimento das estórias focado na entrega das Features;
- Suporte ao Teste Integrado;

As camadas de Gestão Técnica e Desenvolvimento são compostas por Trens, que são um conjunto de *Squads* que podem ser multi sistemas ou mono sistemas.

A composição das *Squads* é sempre de um único fornecedor, como exemplificado na figura 3.1.1.

Não pode ter *Squad* com mais de um fornecedor envolvido, pois os contratos não permitem que os fornecedores participem de *Squads* multidisciplinares com fornecedores diferentes.

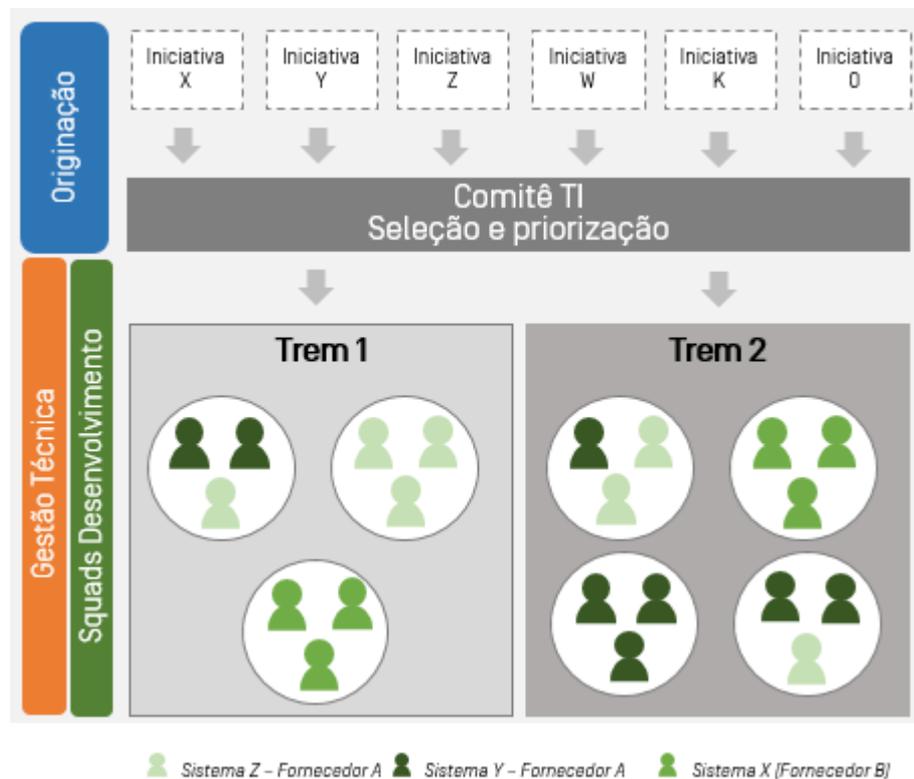


Figura 3.1.1 – Visão Macro da Governança de Projetos Ágeis

Fonte: Autor

Nas figuras 3.1.2, 3.1.3 e 3.1.4 são apresentadas as atividades especificadas de cada camada de gestão.

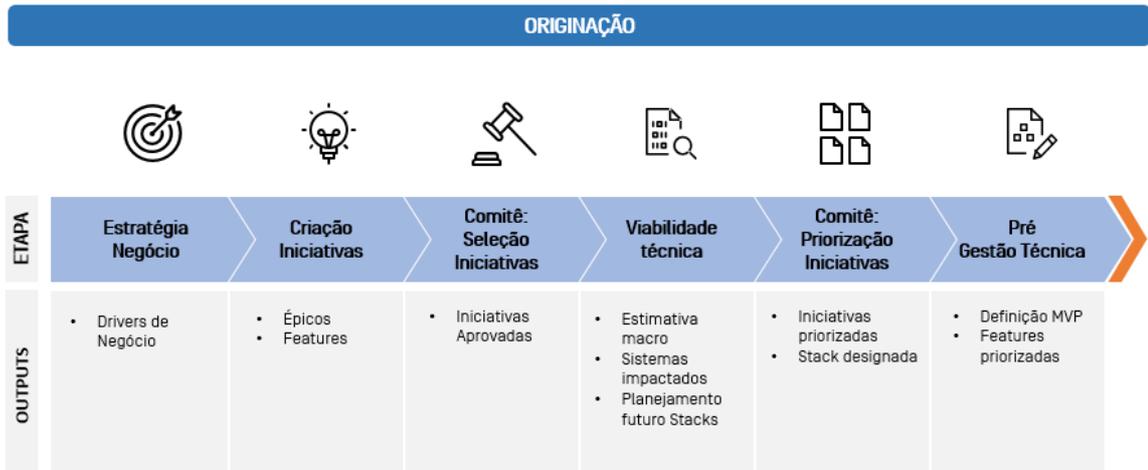


Figura 3.1.2 – Atividades Camada de Gestão – Originação

Fonte: Autor



Figura 3.1.3 – Atividades Camada de Gestão – Gestão Técnica

Fonte: Autor



Figura 3.1.4 – Atividades Camada de Gestão – Desenvolvimento

Fonte: Autor

O quadro 1 apresenta os Eventos e Atividades previstas para acontecerem durante os projetos, considerando as camadas de Gestão Técnica e Desenvolvimento.

Quadro 3.1.1 – Atividades e Participantes dos Eventos e Atividades

	GESTÃO TÉCNICA	DESENVOLVIMENTO
<b>EVENTOS:</b>		
Inspect & Adapt	<b>X</b>	<b>X</b>
Planning Gestão Técnica	<b>X</b>	
Pré BRP	<b>X</b>	<b>X</b>
BRP	<b>X</b>	<b>X</b>
<b>ATIVIDADES:</b>		
Refinamento Macro Features	<b>X</b>	<b>X</b>
Entendimento Estórias		<b>X</b>
Planejamento Macro Sprints	<b>X</b>	<b>X</b>
Mapeamento dependências		<b>X</b>
Mapeamento Riscos		<b>X</b>
Suporte Squads Dev BRP	<b>X</b>	
Suporte a testes		<b>X</b>

Fonte: Autor

Durante o desenho do processo, foram definidas as Squads fixas e as Squads Cross, ou seja, Squads que poderiam atender a uma ou mais Squads de Desenvolvimento.

Os trens foram definidos por produtos e para cada trem tem um conjunto de Squads de sistemas que atendem aquele determinado produto, como exemplificado na figura 3.1.5 um trem do produto Móvel Pré Pago.



Figura 3.1.5 – Exemplo da Composição de um Trem de Produto

Fonte: Autor

E em conjunto ao atendimento desse produto tem as Squads Cross<sup>1</sup>, exemplificado na figura 3.1.6, tendo sua prioridade definida junto no backlog das iniciativas dos Trems.

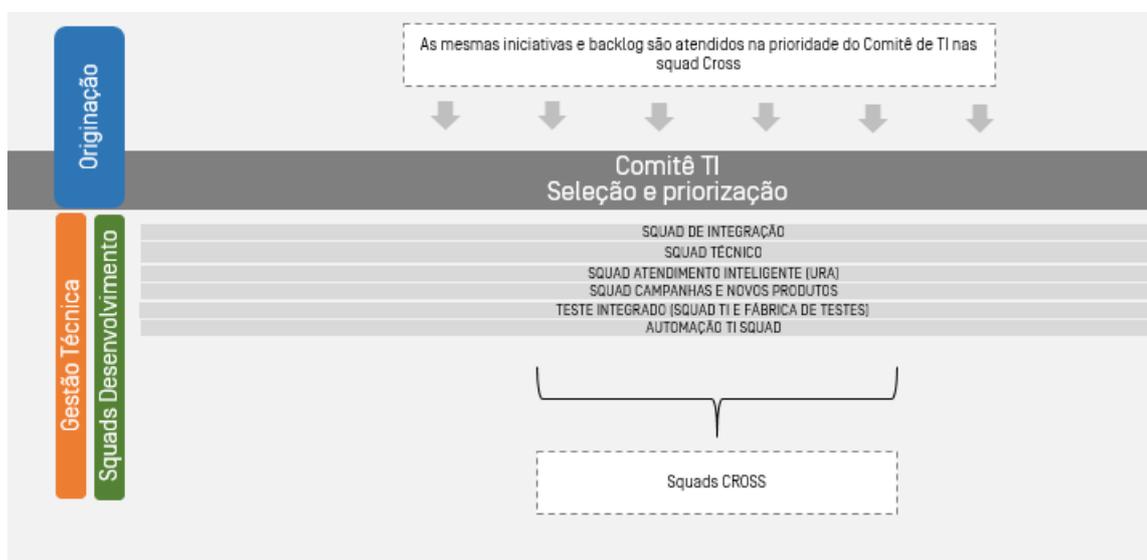


Figura 3.1.6 – Exemplo das Squads Cross

Fonte: Autor

<sup>1</sup> Squad Cross são Squads que atendem a mais de um trem.

### 3.1.1 Papéis e Responsabilidades

Foram redefinidos os papéis e responsabilidade, descritos no quadro 2, para os colaboradores e fornecedores das fábricas de desenvolvimento e fábricas de testes, para que pudessem se adequar aos papéis sugeridos nos métodos ágeis.

Quadro 3.1.1.1 – Papéis e Responsabilidades

PAPEL	DESCRIÇÃO	ÁREA
<p><b>Product Manager</b> (Colaborador)</p>	<p>Tem a função de assegurar o retorno de investimento e benefícios da iniciativa, assim como garantir a entrega do seu Trem em parceria com o RTE. Define e comunica a visão e <i>roadmap</i> da iniciativa e prioriza/reprioriza conforme necessidade. É o responsável por conduzir as agendas de Refinamentos Funcionais junto com o Solicitante, Analista de Negócio e POs, detalhando e, se necessário, apoiando na reescrita das <i>Features</i>.</p>	<p>Gestão de Demandas (Gerente)</p>
<p><b>RTE</b> (Colaborador)</p>	<p>Facilitar os eventos, como a BRP, garantir a governança e processos dos trens e auxiliar as equipes nas entregas de valor. O RTE atua por trem e se comunica com os <i>stakeholders</i>, escala impedimentos, ajuda a gerenciar riscos e impulsiona melhorias. Em conjunto com o PM, o RTE terá a missão de garantir as entregas previstas no seu trem.</p>	<p>Desenvolvimento (executivo)</p>
<p><b>Solicitante</b> (Colaborador)</p>	<p>São as pessoas que representam as áreas clientes e têm a função de criar ideias e estratégias de acordo com o cenário da companhia, defender a necessidade, escrever iniciativas e <i>Features</i>, suportar os POs sobre as dúvidas funcionais e acompanhar o desenvolvimento das iniciativas (<i>Features</i> e USs).</p>	<p>Áreas Clientes (Negócio, TI, Operações)</p>
<p><b>Analista de Negócio</b> (Colaborador)</p>	<p>Responsável pela gestão da etapa de Originação junto a área cliente e por analisar e sintetizar</p>	<p>Gestão de Demandas e TI</p>

	informações de negócios, criando uma conexão entre a área cliente e TI.	
<b>Arquiteto de Soluções</b> (Colaborador)	Define e compartilha visões técnicas e de arquitetura dos sistemas. Responsável pela elaboração do desenho da solução na etapa de viabilidade técnica na camada de Originação.	Arquitetura de Soluções
<b>Analista do Trem</b> (Colaborador)	Apoia o PM e PO na reunião de Refinamento Funcional e Técnico do trem detalhando e, se necessário, apoiando na reescrita das <i>Features</i> . Também suporta os POs e <i>Storywriters</i> na escrita e refinamento funcional das USs. Esse perfil atua por trem e tem a responsabilidade de disponibilizar a visão do <i>backlog</i> do trem.	Gestão de Demandas
<b>Arquiteto de Sistema</b> (Colaborador ou Fornecedor)	Participa das reuniões de refinamento funcional e técnico das <i>Features</i> e histórias e suporta as <i>Squads</i> de desenvolvimento. Esse perfil tem uma atuação por Sistema.	Desenvolvimento (São especialistas dos sistemas)
<b>Responsável Técnico</b> (Colaborador)	Atuação por sistema, e participação na camada de Gestão Técnica como apoio no Refinamento Funcional e Técnico, e na camada de <i>Squads</i> de Desenvolvimento como suporte ao DEV TEAM na solução do projeto e na elaboração dos testes (TU e TS) referentes aos sistemas da área em que atua. Além disso, esse papel suporta e acompanha, os testes integrados, UAT, entrada em produção e pós-produção da área impactada.	Desenvolvimento
<b>PO</b> (Colaborador)	Define, escreve e aceita as histórias e também atua como ponto focal de Negócio para esclarecer as dúvidas dos desenvolvedores. O PO tem uma atuação por <i>Squad</i> e trabalha no gerenciamento dos ciclos da BRP para garantir os acordos e prioridades alinhadas com PM/RTE.	Áreas de Negócio
<b>Test Lead</b> (Colaborador)	Responsável pelo entendimento da demanda que precisa de Teste Integrado, classifica se a demanda será atendida na <i>Squad</i> de TI ou	Qualidade/Testes

	contratação fábrica, internaliza e organiza as demandas no <i>backlog</i> de Testes.	
<b>Facilitador Teste</b> (Colaborador)	Responsável pelo acompanhamento do planejamento e da execução dos testes, priorização dos defeitos e report do andamento dos testes.	Qualidade/Testes
<b>Time de Teste</b> (Fornecedor – Fábrica de Teste)	Responsável pelo planejamento e execução dos testes automatizados e manuais.	Qualidade/Testes
<b>Analista de Automação</b> (Colaborador)	Responsável pelo entendimento da demanda que precisa da automação dos testes, internaliza e organiza as demandas no backlog de Automação.	Qualidade/Garantia da Qualidade
<b>Time de Automação</b> (Fornecedor – Fábrica de Teste)	Responsável pela escrita das USs referente a automação necessária dos testes, desenvolve e testa (teste unitário) o script criado / corrigido.	Qualidade/Garantia da Qualidade
<b>Storywriter</b> (Colaborador ou Fornecedor)	Suporta a escrita e o refinamento técnico das histórias. Também atua como ponto focal para esclarecer as dúvidas dos desenvolvedores. O <i>storywriter</i> tem uma atuação por <i>Squad</i> e trabalha de forma consultiva com o PO no planejamento dos ciclos da BRP.	Áreas de Negócio Qualidade/Testes
<b>Dev Team</b> (Fornecedor – Fábrica de Desenvolvimento)	Time de 3 a 9 membros, responsável pela estimativa das USs e que apoia no refinamento das histórias de usuários e critérios de aceitação. Desenvolve e compromete-se com os objetivos definidos na BRP e os planos de entregas por <i>Sprint</i> . Os desenvolvedores também são responsáveis por realizar os testes de sistemas <sup>2</sup> das histórias e <i>Features</i> entregues e futuramente suportar os testes integrados.	Desenvolvimento / Fornecedor
<b>PM</b> (Colaborador)	Responsável por estabelecer as melhores práticas e diretrizes de utilização dos sistemas e gerenciar	Desenvolvimento

<sup>2</sup> Teste de Sistemas na ótica da companhia são os testes realizados nas fronteiras do próprio sistema sem nenhuma integração com outros sistemas.

	as licenças dos mesmos, assim como pela gestão do fornecedor dos seus sistemas.	
<b>Scrum Master</b> (Colaborador)	Líder servidor e <i>coach</i> para a equipe, inculindo o processo de agilidade acordado, removendo impedimentos e promovendo um ambiente de alto desempenho, fluxo contínuo e melhorias.	Desenvolvimento

Fonte: Autor

### 3.1.2 Ritos e Cerimônias

Também de acordo com os métodos ágeis que foram descritos no capítulo 2 deste trabalho, durante a reestruturação do processo foram definidos os ritos e cerimônias previstas para ocorrer em cada uma das camadas de gestão, qual a periodicidade que acontecem as cerimônias, qual o nível (Iniciativa, Trem ou *Squad*) e quem são os participantes de cada uma delas.

Quadro 3.1.2.1 – Ritos e Cerimônias – Camada de Originação

ORIGINAÇÃO				
NOME	DESCRIÇÃO	PERIODO	NÍVEL	PARTICIPANTES
<b>Reunião Comitê TI</b>	Apresentação de novas iniciativas e seleção das que irão para a etapa de viabilidade técnica	Mensal	Iniciativas	<ul style="list-style-type: none"> <li>• Diretores</li> <li>• Solicitantes</li> <li>• PMs</li> <li>• RTEs</li> </ul>
<b>Reunião Comitê de Gestão (CDG)</b>	Priorização das iniciativas aprovadas e alocação das mesmas no pipeline dos trens	Mensal	Iniciativas	<ul style="list-style-type: none"> <li>• Presidente</li> <li>• Diretores</li> <li>• PMs</li> <li>• RTEs</li> </ul>
<b>Backlog pós CDG</b>	Avaliação do <i>backlog</i> do Trem após a priorização das iniciativas no CDG e alocação das mesmas nos Trens. Planejado o <i>backlog</i> dos trens com previsão de quando as iniciativas serão tratadas na Gestão Técnica, para	Mensal	Iniciativas	<ul style="list-style-type: none"> <li>• RTE</li> <li>• Solicitantes</li> <li>• PMs</li> <li>• Analista de Negócio</li> <li>• Analista do Trem</li> </ul>

	seguir para as <i>Squads</i> de Desenvolvimento.			
<b>Checkpoint artefatos pré gestão técnica</b>	Reunião para garantir que as <i>Features</i> das iniciativas priorizadas foram devidamente organizadas por ciclos e estão ok para passarem para a camada de gestão técnica. O RTE é a pessoa que puxa a reunião e os Analistas de Negócio / Solicitante apresentam os artefatos organizados.	Mensal	Iniciativas	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Solicitante</li> <li>• Analista de Negócio</li> </ul>

Fonte: Autor

Quadro 3.1.2.2 – Ritos e Cerimônias – Camada Técnica

GESTÃO TÉCNICA				
NOME	DESCRIÇÃO	PERIODO	NÍVEL	PARTICIPANTES
<b><i>Planning GT</i></b>	Planejamento macro das atividades da camada de gestão técnica para o próximo ciclo de 3 <i>Sprints</i>	1x por ciclo da <i>Sprint</i>	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Analista do Trem</li> <li>• Solicitantes</li> <li>• Analistas de Negócio</li> <li>• Arquitetos de Sistema</li> <li>• Responsáveis Técnicos</li> <li>• POs</li> <li>• Storywriters</li> </ul>
<b><i>Planning Sprint GT</i></b>	Refina o planejamento realizado na <i>Planning GT</i> , determina as metas da	1x por <i>Sprint</i> (dia 1 da <i>Sprint</i> )	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> </ul>

	<i>Sprint</i> em questão e <i>backlog</i> das próximas <i>Sprints</i> .			<ul style="list-style-type: none"> <li>• Analista do Trem</li> <li>• Solicitantes</li> <li>• Analistas de Negócio</li> <li>• Arquitetos de Sistema</li> <li>• Responsáveis Técnicos</li> <li>• POs</li> <li>• Storywriters</li> <li>• Test Lead</li> <li>• Analista de Automação</li> </ul>
<b>Refinamento Funcional e Técnico</b>	Reunião para esclarecimento de eventuais dúvidas e detalhamento funcional e técnico das <i>Features</i> . Se necessário as <i>Features</i> poderão ser reescritas e até “ <i>splitadas</i> ” nesse evento.	1x por <i>Sprint</i> por iniciativa	Trem	<ul style="list-style-type: none"> <li>• PM</li> <li>• Analista do Trem</li> <li>• Solicitantes</li> <li>• Analistas de negócio</li> <li>• POs</li> <li>• Storywriters</li> <li>• Arquitetos de Sistemas</li> <li>• Responsáveis técnicos</li> <li>• RTE</li> <li>• Gestor Técnico</li> <li>• Test Lead</li> <li>• Analista de Automação</li> </ul>
<b>Refinamento USs</b>	Após a escrita das USs e refinamento das <i>Features</i> , acontece a reunião para <b>refinar as estórias</b> .	1x por <i>Sprint</i> por iniciativa	Iniciativa	<ul style="list-style-type: none"> <li>• Solicitante</li> <li>• Analista do Trem</li> <li>• PM</li> </ul>

				<ul style="list-style-type: none"> <li>• Analista de negócio</li> <li>• POs</li> <li>• Storywriters</li> <li>• Responsáveis Técnicos</li> <li>• Arquiteto de Sistemas</li> <li>• Test Lead</li> <li>• Analista de Automação</li> </ul>
<b>Review / Retrospectiva Sprint GT</b>	<p><u>Review</u>: Time analisa os resultados entregues na <i>Sprint</i> em questão e conforme necessidade ajusta o <i>backlog</i> para as <i>Sprints</i> seguintes.</p> <p><u>Retrospectiva</u>: Time revisa suas práticas e identifica maneiras de melhorar o processo como um todo</p>	1 x por <i>Sprint</i> (último dia da <i>Sprint</i> )	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Analista do Trem</li> <li>• Solicitantes</li> <li>• Analistas de negócio</li> <li>• Arquitetos de sistema</li> <li>• Responsáveis técnicos</li> <li>• POs</li> <li>• Storywriters</li> <li>• Test Lead</li> </ul>

Fonte: Autor

Quadro 3.1.2.3 – Ritos e Cerimônias – Camada Desenvolvimento

DESENVOLVIMENTO				
NOME	DESCRIÇÃO	PERÍODO	NÍVEL	PARTICIPANTES
<b>Big Room Planning</b>	Evento que garante o entendimento e alinhamento entre todos os times (negócio e técnico) sobre o que será realizado no próximo	1x a cada ciclo no <i>Sprint</i>	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Solicitantes</li> <li>• Analista do Trem</li> </ul>

	<p>ciclo (3 <i>Sprints</i>) distribuição <i>Features/USs x Sprint</i>, mapeamento de riscos e alinhamento de dependências.</p>			<ul style="list-style-type: none"> <li>• Analistas de negócio</li> <li>• Arquitetos de Sistemas</li> <li>• Responsáveis Técnicos</li> <li>• POs</li> <li>• Storywriters</li> <li>• Times Dev</li> <li>• Test Lead</li> <li>• Gestor Técnico</li> <li>• Arquitetos de Solução</li> <li>• Time de Testes</li> <li>• Analista Automação</li> </ul>
<b><i>Inspect &amp; Adapt</i></b>	<p>Evento composto por três etapas (System Demo, métricas e retro) que tem como objetivo incentivar reflexão sobre o que foi realizado até o momento e criar um plano de ação de melhorias para o próximo ciclo.</p>	1x a cada ciclo no <i>Sprint</i>	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Solicitantes</li> <li>• Analista do Trem</li> <li>• Analistas de negócio</li> <li>• Arquitetos de Sistemas</li> <li>• Responsáveis Técnicos</li> <li>• RTEs</li> <li>• POs</li> <li>• Storywriters</li> <li>• SM</li> <li>• Dev Team</li> <li>• Test Lead</li> <li>• Arquitetos de Solução</li> </ul>

				<ul style="list-style-type: none"> <li>• Analista Automação</li> <li>• Facilitador Teste</li> </ul>
<b>Planning Sprint Squad</b>	Reunião de refinamento do planejamento realizado na BRP, onde o time determinará as metas da <i>Sprint</i> em questão e quanto do <i>backlog</i> eles podem se comprometer durante as próximas <i>Sprints</i> .	1x por <i>Sprint</i> (Dia 1 da <i>Sprint</i> )	<i>Squad</i>	<ul style="list-style-type: none"> <li>• SM</li> <li>• PO</li> <li>• Storywriter</li> <li>• Dev Team</li> <li>• RTE</li> <li>• Arquiteto de Sistema</li> <li>• Responsável Técnico</li> </ul>
<b>Daily</b>	Evento que tem como objetivo dar clareza sobre o que foi feito no dia anterior, identifica impedimentos e prioriza o trabalho a ser realizado no dia em questão	Diariamente	<i>Squad</i>	<ul style="list-style-type: none"> <li>• SM</li> <li>• PO</li> <li>• Storywriter</li> <li>• Dev Team</li> <li>• Arquiteto de Sistema</li> <li>• Responsável Técnico</li> </ul>
<b>Sincronismo</b>	Reunião semanal que ajuda a coordenar as dependências das <i>Squads</i> e fornece visibilidade do progresso e impedimentos.	1x por semana	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Analista de trem</li> <li>• RTEs</li> <li>• Solicitantes</li> <li>• POs</li> <li>• Test Lead</li> <li>• Analista de Automação</li> <li>• Facilitador Teste</li> <li>• Analista de Negócio</li> </ul>

				<ul style="list-style-type: none"> <li>• Arquitetos de Sistemas</li> <li>• Responsáveis Técnicos</li> </ul>
<b>Review / Retro Sprint</b>	<p><u>Review</u>: Time analisa os resultados entregues na <i>Sprint</i> em questão e conforme necessidade ajusta o <i>backlog</i> para as <i>Sprints</i> seguintes.</p> <p><u>Retro</u>: Time revisa suas práticas e identifica maneiras de melhorar o processo como um todo.</p>	1x por <i>Sprint</i> (Último dia da <i>Sprint</i> )	<i>Squad</i>	<ul style="list-style-type: none"> <li>• SM</li> <li>• PO</li> <li>• Storywriter</li> <li>• Dev Team</li> <li>• RTE</li> <li>• Arquiteto de Sistema</li> <li>• Responsável Técnico</li> </ul>
<b>Alinhamento BRP</b>	Reunião de alinhamento e explicação sobre a agenda, dinâmicas e ferramentas que serão utilizadas na BRP.	1x a cada ciclo no <i>Sprint</i>	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Solicitantes</li> <li>• POs</li> <li>• Analista do Trem</li> <li>• Test Lead</li> <li>• Analista de Negócio</li> </ul>
<b>Pré BRP</b>	Evento onde os times se reúnem para alinhar se tudo está devidamente preparado para a realização da BRP	1x a cada ciclo no <i>Sprint</i>	Trem	<ul style="list-style-type: none"> <li>• RTE</li> <li>• PM</li> <li>• Solicitantes</li> <li>• POs</li> <li>• Analista do Trem</li> <li>• SMs</li> <li>• Test Lead</li> <li>• Analista de Automação</li> <li>• Analista de Negócio</li> </ul>

Fonte: Autor

## 3.2 Processo da área de Gestão de Testes

A área de Gestão de Testes, é responsável por todos os testes integrados<sup>3</sup>, testes de homologação (UAT) e testes de regressão<sup>4</sup> dos sistemas críticos da companhia.

Atualmente a companhia tem em torno de 100 sistemas na sua operação. Desses, 30 são escopo de atualização da área de Gestão de Testes. Em média a área de testes entrega cerca de 40 projetos por release e tem como principais atividades:

- Desenvolver a estratégia de testes dos projetos levantando todos os Cenários e Casos de Testes pertinentes;
- Realizar a contratação financeira dos testes dos projetos junto aos respectivos fornecedores de testes;
- Coordenar os fornecedores durante o planejamento dos testes, garantindo que todos os cenários e casos de testes especificados em tempo de Estratégia de Testes estejam planejados e aprovados pelos respectivos responsáveis (área técnica e área usuária);
- Planejar e conduzir reuniões de interdependência de cenários de testes entre fornecedores/sistemas diferentes, gerando o plano de testes consolidado como produto final;
- Participar ativamente dos testes integrados e testes de homologação referentes aos projetos, garantindo os padrões de qualidade;
- Gerenciar os testes com uma abordagem fim a fim e com foco nas entregas;
- Coordenar, alinhar, registrar, acompanhar a execução dos testes sempre alinhado com as diversas equipes técnicas envolvidas nos projetos;
- Gerar status diário da execução dos testes, apresentando como está a evolução dos testes e os defeitos identificados no decorrer dos testes, e como está o tratamento desses defeitos;
- Participação ativa junto a equipe de release, que é responsável por gerir o andamento dos projetos que estão previstos para entradas definidas em produção.

---

<sup>3</sup> Testes Integrados são testes realizados entre vários sistemas que compõe uma funcionalidade.

<sup>4</sup> Testes de regressão são realizados para garantir que o código novo que irá subir para produção não irá gerar nenhum impacto naquilo que já está sendo executado.

Na figura 3.2.1, são apresentadas as principais atividades da Gerência de Qualidade e suas áreas:



Figura 3.2.1 – Principais atividades da Gerência de Qualidade e suas áreas

Fonte: Autor

A Gerência de Qualidade tem diversas áreas que trabalham em conjunto para garantir a qualidade nas entregas dos projetos desenvolvidos pela TI da companhia.

Atualmente toda TI trabalha com diferentes fornecedores contratados para a realização do desenvolvimento e para a realização do planejamento e execução dos testes. Os colaboradores da empresa são responsáveis por realizar a gestão dessas atividades junto aos fornecedores. Porém o processo da TI é definido internamente na companhia e seus fornecedores precisam seguir esse processo, o que algumas vezes vão contra cláusulas contratuais e é necessário “adaptar” o processo para que o contrato seja atendido.

O modelo de contratação dos fornecedores prevê um volume de horas por mês, utilizadas para atendimento aos projetos, essas horas são geridas da seguinte forma:

- Baseline – São horas previstas de capacidade de atendimento aos projetos com taxa pré-definida em contrato e volume de horas mensais;
- Extra Baseline – Utilizado quando as horas do baseline são extrapoladas, porém tem um custo maior da taxa horária. Este modelo de contratação deve ser utilizado somente em casos extremos, quando tem a necessidade de implantar um projeto

emergencial e não pode aguardar disponibilidade de baseline para atendimento (Exemplo: Projeto regulatórios, correção de erros críticos em produção).

Para cada projeto que a área de testes é impactada, é necessário a elaboração de um documento de Estratégia de Testes, que contém o escopo dos testes que precisam ser realizados, como: Quais os sistemas impactados; cenários e casos de Testes que deverão ser executados; macro cronograma de atendimento; em qual ambiente os testes serão executados; qual o modelo de homologação será realizado (por evidência ou por execução do Cliente/Usuário). Este documento de estratégia é aprovado pelos envolvidos no projeto, tanto da área técnica (TI) quanto das áreas usuárias (áreas de negócio) e quando o projeto é tratado pelo modelo Waterfall, este documento serve de insumo para que o fornecedor gere sua proposta técnica e financeira para atender ao projeto.

Tanto no modelo Waterfall como no modelo ágil, o documento de Estratégia de Testes é utilizado como insumo para a etapa de Planejamento dos Testes, etapa onde são desenhados os Cenários e Casos de Testes na ferramenta de apoio à execução dos testes.

No modelo Waterfall as atividades de Planejamento e Execução dos testes só são iniciadas após a aprovação financeira formal de cada proposta. As aprovações acontecem uma vez por semana, durante um comitê executivo, onde são apresentadas e justificadas as propostas. Este processo gera muita demora no início das atividades, pois enquanto as aprovações formais não são emitidas, os fornecedores não iniciam o Planejamento e/ou a Execução dos testes.

Outros tipos de testes como: Testes Unitários, Testes de Sistemas, Testes de Performance e Testes de Segurança não são escopo da área de testes. São testes realizados pelos próprios fornecedores de desenvolvimento.

A área de testes trabalha com atividades sequenciais onde uma atividade só começa quando a anterior termina e essas atividades são desenvolvidas por equipes diferentes dentro da área, conforme demonstra a figura 3.2.2.

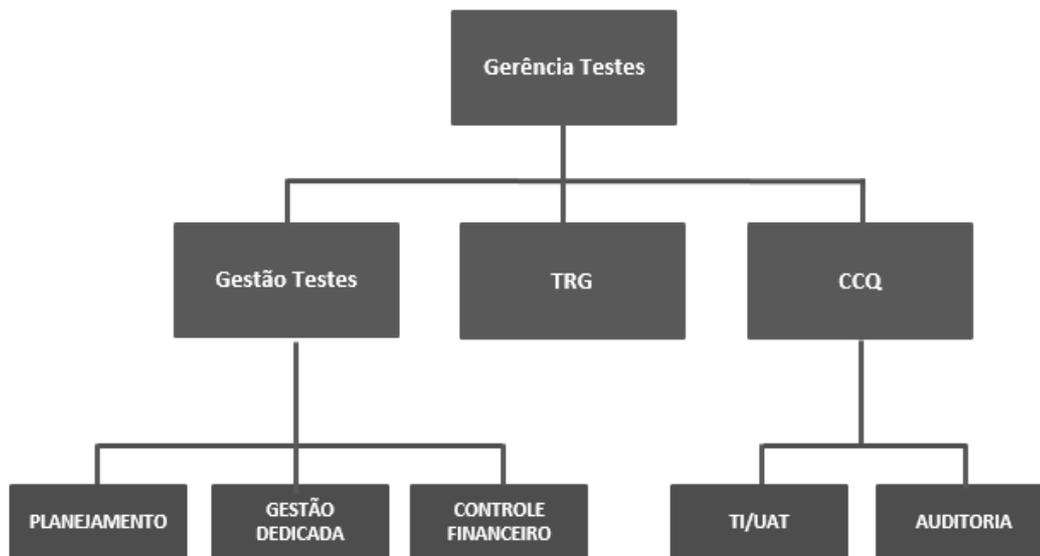


Figura 3.2.2 – Estrutura da Área de Gestão de Testes

Fonte: Autor

Detalhamento das atividades desempenhadas em cada uma das equipes da área:

**Gestão de Testes** – Equipe responsável pelo processo inicial da área. Quando um projeto é demandado para a área de testes, ele é recepcionado pela equipe de Gestão de Testes, que faz a primeira análise de escopo e verifica se os sistemas impactados são atendidos pela área. Uma vez identificado que o projeto atende as características de atuação da área, é designado um líder de testes que irá acompanhar o Planejamento dos Testes;

- Planejamento – Uma vez identificado que o projeto atende o escopo da área de testes, é designado um líder de testes que irá realizar o planejamento dos testes. O líder de testes analisa e valida os requisitos dos projetos e os critérios de aceitação. Elabora o documento de Estratégia de Testes com a todas as informações necessárias para a execução dos testes. Realiza a aprovação financeira das propostas da fábrica de testes. Valida e aprova o cronograma de acordo com a expectativa de implantação do projeto. Acompanha as atividades de elaboração do Plano de Teste Integrado e UAT na ferramenta de Apoio a Execução dos Testes. Quando finalizado o plano de teste é aprovado pelos responsáveis técnicos e pelos usuários solicitantes. Com o plano de testes aprovado o líder de teste de planejamento faz um Handover para o líder de testes da equipe do CCQ, para o acompanhamento da execução dos testes.
- Gestão Dedicada – Essa equipe faz toda a atividade da área de planejamento e execução dos testes, geralmente gestão dedicada acontece para projetos críticos para

a companhia, como projetos estruturantes, regulatórios ou que tenham um cronograma muito agressivo para atender alguma demanda crítica de mercado, como datas especiais (Dia das mães e Natal, por exemplo);

- Controle Financeiro – Nessa etapa é realizado o controle financeiro da área.

**TRG** – Teste de Regressão Geral, etapa onde são realizados testes em ambiente de pré-produção com o objetivo de garantir que os códigos dos projetos que subirão numa determinada Release não irão gerar nenhum impacto nas funcionalidades já existentes em produção. Estes testes são os últimos na cadeia de testes;

**CCQ** – Centro de Controle de Qualidade, área responsável pelo acompanhamento da execução dos testes. Os testes são executados pela fábrica de testes e a equipe do CCQ acompanha o andamento, fazendo report diário da evolução dos testes, acompanha quando algum defeito é encontrado até a sua solução, acompanha a liberação das evidências de teste que devem ser aprovadas pelos responsáveis técnicos dos sistemas e pelos usuários solicitantes. Nessa etapa são acompanhados os Testes Integrados e Testes de Homologação (UAT – User Acceptance Test), além de ser a equipe responsável pelas auditorias de defeitos;

- TI/UAT – É propriamente a atividade acompanhamento da execução dos Testes Integrados (TI) e UAT (Teste de Aceitação do Usuário). Uma vez o planejamento concluído e aprovado a equipe do CCQ assume a responsabilidade pelos testes do projeto.
- Auditoria – Etapa realizada após a conclusão dos testes. É realizada uma reunião com todos os envolvidos no projeto (fábrica de testes, fábricas de desenvolvimento, responsáveis técnicos dos sistemas, equipe de gestão de ambiente não produtivo) e são analisados os defeitos identificados, as naturezas dos defeitos e os agentes ofensores. Todos devem estar de comum acordo com o encaminhamento do defeito, pois são analisados em indicadores contratuais podendo gerar multa para os respectivos responsáveis.

### **3.2.1 Resumo de Eventos com Insumo para Área de Testes**

Conforme exposto no item 3.1.2 foram definidas cerimônias que irão apoiar todo o ciclo dos projetos. Durante essas cerimônias é possível que atividades iniciem em paralelo para dar andamento aos projetos e isso acontece fortemente para atividades da área de testes, quando ela atua nos projetos ágeis.

Abaixo, as figuras 3.2.1.1, 3.2.1.2, representam os eventos que são insumos para que a equipe de teste inicie as suas atividades.

Ciclo [Camada de Gestão Técnica]							
Sprint 1		Sprint 2		Sprint 3		Sprint R&P	
Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8
1 Refinamento Funcional	Escrita US	Refinamento Funcional	Escrita US	Refinamento Funcional	Escrita US	2 Evento Inspect & Adapt	3 Planejamento Gestão Técnica próximo ciclo
Refinamento Técnico	Refinamento US	Refinamento Técnico	Refinamento US	Refinamento Técnico	Refinamento US	Suporte Squads Dev refinamento e planejamento BRP	Evento BRP

Figura 3.2.1.1 – Eventos no Ciclo da Camada Técnica

Fonte: Autor

**Refinamento Funcional:** Evento em que o Test Lead participa para iniciar a elaboração do documento de Estratégia de Testes;

**Escrita US:** Na última semana da *Sprint 3* da camada técnica é finalizada a escrita e refinamento das USs que foram priorizadas no evento de BRP. Com as USs refinadas o Test Lead finaliza a elaboração do documento da Estratégia de Testes;

**Evento Inspect & Adapt:** Na semana que acontece este evento, onde é avaliado o que foi feito até este momento e identificar pontos de melhoria para o próximo ciclo. Nesta semana ocorre reunião de validação da Estratégia de Testes e é dado início ao processo de Planejamento e Execução dos Testes.

Ciclo [Camada Desenvolvimento]							
Sprint 1		Sprint 2		Sprint 3		Sprint R&P	
Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8
1 Desenvolvimento	Desenvolvimento	Desenvolvimento	Desenvolvimento	Desenvolvimento	Desenvolvimento	Evento Inspect & Adapt	Refinamento e planejamento Pré BRP
Desenvolvimento	3 Suporte teste integrado e automação	Desenvolvimento	Suporte teste integrado e automação	Desenvolvimento	Suporte teste integrado e automação	Refinamento e planejamento Pré BRP	Evento BRP
2 Automatização e Execução dos Testes	Automatização e Execução dos Testes	Automatização e Execução dos Testes	Automatização e Execução dos Testes	Automatização e Execução dos Testes	Automatização e Execução dos Testes	Automatização e Execução dos Testes	

Figura 3.2.1.2 – Eventos no Ciclo da Camada de Desenvolvimento

Fonte: Autor

**Desenvolvimento:** Enquanto inicia o desenvolvimento no *Sprint 1*, a fábrica de testes está elaborando o Plano de Teste que será testado no próximo ciclo;

**Automatização e Execução dos Testes:** Neste momento estão sendo executados os testes que foram definidos no ciclo anterior;

**Suporte Teste Integrado e Automação:** O fornecedor de desenvolvimento tem como atividade o suporte aos Testes Integrados e as Automações. Durante a execução dos testes os defeitos encontrados são tratados pela equipe de desenvolvimento (quando se trata de defeito de código), isso é caracterizado como Suporte aos Testes.

De acordo com o estudo realizado no capítulo 2, foi possível observar que o processo que está sendo implantado, não está seguindo em sua totalidade os conceitos e princípios do *SAFe* e dos demais métodos que foram expostos e que a área de testes precisa de uma reestruturação que seja capaz de modificar seu modo de atuação, principalmente de atuar com um único modelo de trabalho. Atuar com dois modelos de projeto gera falta de controle, dificuldade na comunicação, demora nas entregas e certamente acarreta numa qualidade abaixo do esperado.

# Capítulo 4

## Resultados Obtidos

### 4.1 –Análise de Tempo de Entrega do Planejamento de Testes (Waterfall x Ágil)

Durante o trabalho de análise do processo da Área de Testes foi identificado que não foram estabelecidos indicadores para medir as melhorias que este trabalho traz para a companhia, como por exemplo o tempo de planejamento dos testes no modelo anterior (Waterfall) e no novo (Ágil). Embora o processo da área ainda esteja em modelo híbrido, como foi apresentado no capítulo anterior, entende-se que seria importante realizar este tipo de análise, até mesmo para que seja possível maior patrocínio dos Stackholders nas mudanças propostas para a área, inclusive para que seja possível o engajamento de todos para transformar toda a atuação da área em modelo ágil.

Diante da situação de não haver dados registrados de forma compartilhada para análise dessas informações, foi realizada uma pesquisa com algumas pessoas do time de testes que participaram de projetos no modelo Waterfall no ano de 2019 (antes da mudança do processo) e participaram de projetos em 2020 no modelo ágil.

Para esta pesquisa foi utilizada como premissa a análise de projetos com as mesmas características, como:

- Sistemas impactados compõem o escopo da Squad de Testes;
- Quantidade de sistemas envolvidos;
- Complexidade das funcionalidades;
- Quantidade de casos de testes.

Foram analisados 10 (dez) projetos de 2019 e 10 (dez) projetos de 2020 e foram analisadas duas etapas importantes da fase de Planejamento dos Testes, como:

- Tempo de Elaboração e Aprovação da Estratégia de Testes;
- Tempo de Elaboração e Aprovação do Planejamento dos Casos de Testes.

Nas figuras 4.1.1, 4.1.2, 4.1.3 e 4.1.4 são apresentados os gráficos com os resultados obtidos dessa análise e na sequência o ganho que foi observado com a implantação da squad de testes.

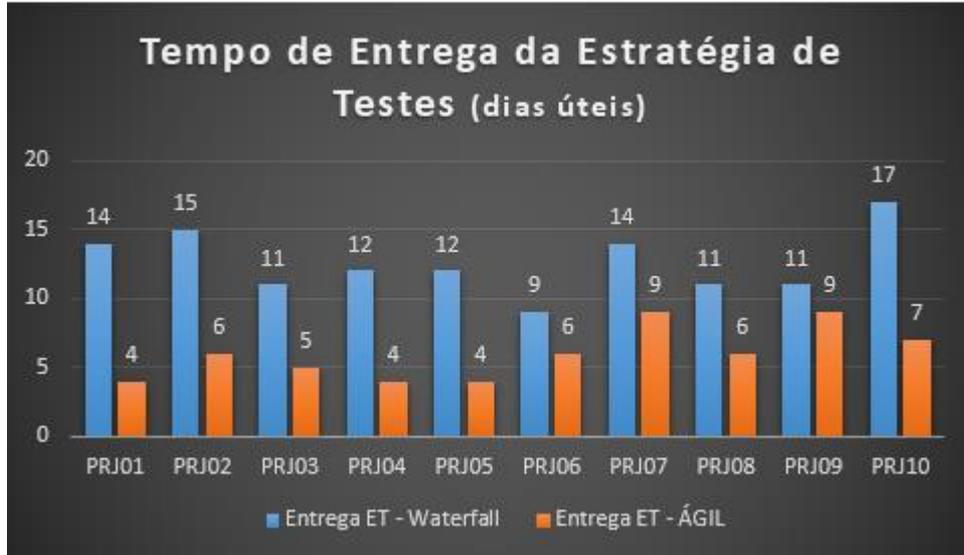


Figura 4.1.1 – Gráfico do Tempo de Entrega da Estratégia de Testes (Waterfall x Ágil)

Fonte: Autor

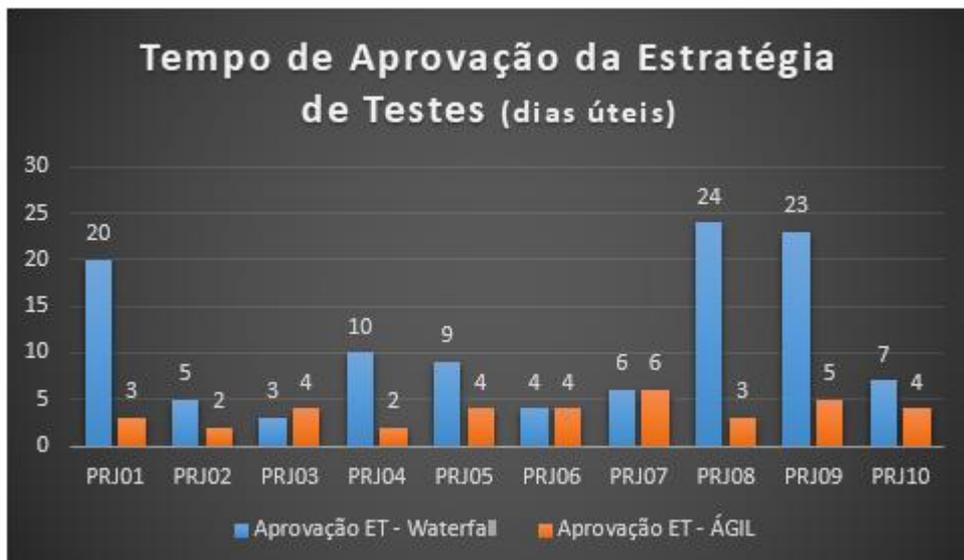


Figura 4.1.2 – Gráfico do Tempo de Aprovação da Estratégia de Testes (Waterfall x Ágil)

Fonte: Autor

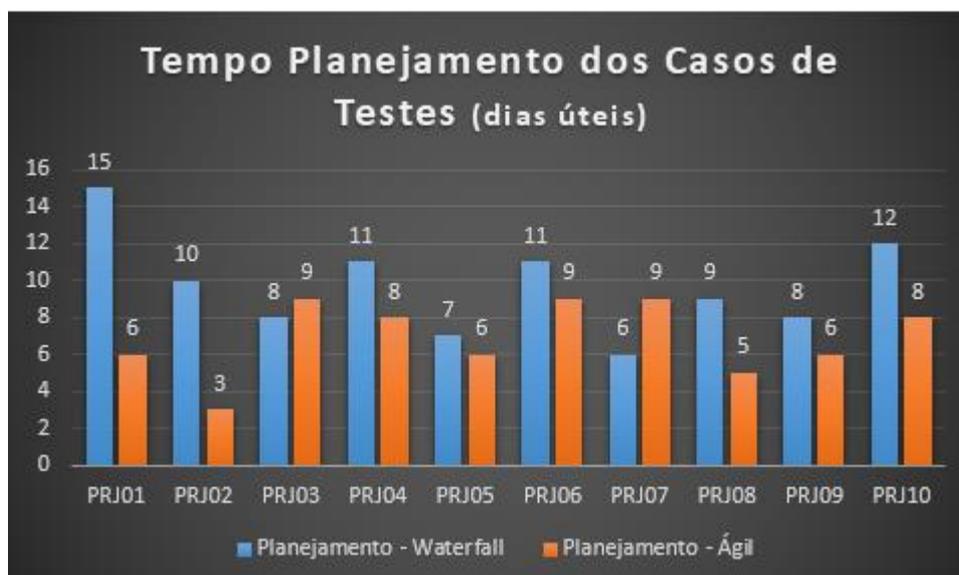


Figura 4.1.3 – Gráfico do Tempo de Planejamento dos Casos de Testes (Waterfall x Ágil)

Fonte: Autor

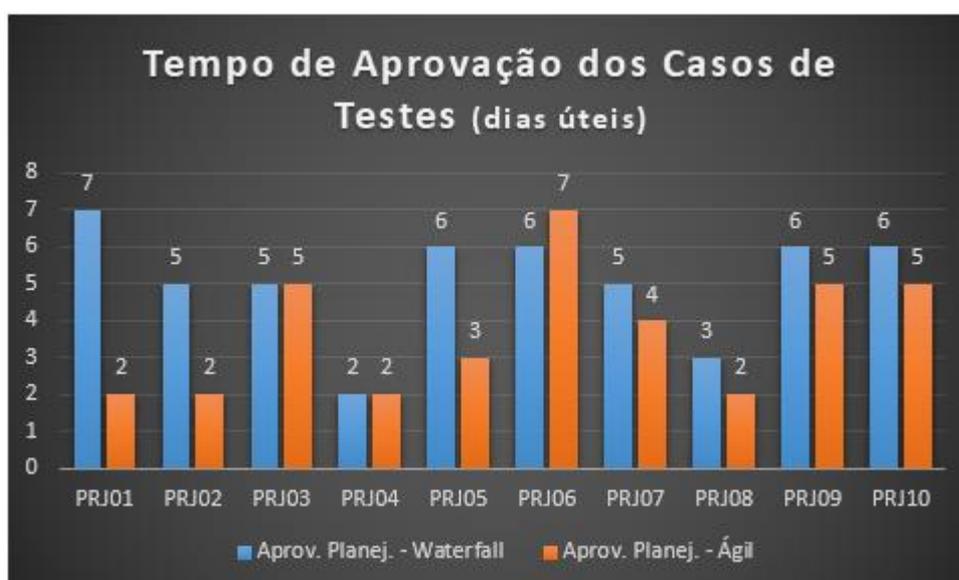


Figura 4.1.4 – Gráfico do Tempo de Aprovação dos Casos de Testes (Waterfall x Ágil)

Fonte: Autor

Observa-se nesses gráficos que na maioria dos projetos o tempo de entrega no processo Waterfall é maior que o tempo de entrega no processo Ágil.

Através dessa pesquisa foi possível identificar os ganhos iniciais com a mudança do processo. Na média houve uma redução de aproximadamente 20 dias úteis no tempo de entrega do Planejamento dos Testes nos projetos, conforme apresentado na tabela 4.1.1.

Tabela 4.1.1 – Tempo Médio de Planejamento dos Testes

<b>Atividades</b>	<b>Média de dias no processo Waterfall</b>	<b>Média de dias no Processo Ágil</b>
Média de entrega de uma estratégia de testes	12,6	6
Tempo médio de aprovação de ET	11,1	3,7
Tempo médio de aprovação de PT	2,8	N/A
Tempo médio para recebimento aprovação financeira da proposta	6,9	N/A
Tempo médio para planejamento dos testes	9,7	6,9
Tempo médio para aprovação do plano de testes	5,1	3,7
<b>TEMPO MÉDIO DE PLANEJAMENTO DE TESTES</b>	<b>48,2</b>	<b>20,3</b>

Fonte: Autor

Um ponto importante para ser observado é que com o processo ágil o tempo foi reduzido também em função de não haver mais a necessidade de geração de proposta técnica e aprovação financeira da mesma. Com a implantação da Squad de Testes, esta etapa foi eliminada pois o custo dos testes já está embutido na capacidade de entrega da Squad. A figura 4.1.5 apresenta o tempo que foi gasto nos projetos de 2019 no modelo Waterfall que contem essa etapa.

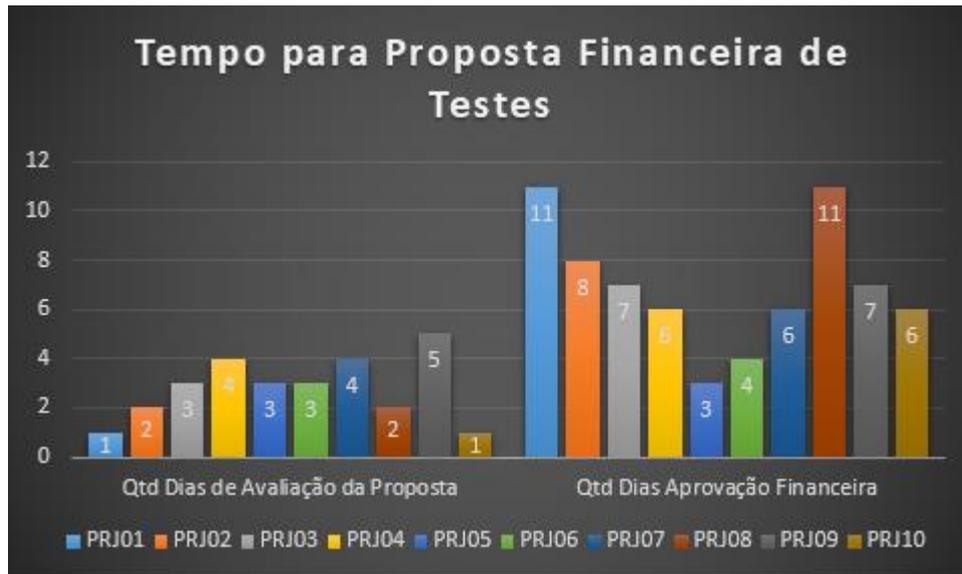


Figura 4.1.5 – Gráfico do Tempo para Aprovação de Proposta Financeira (Waterfall)

Fonte: Autor

## 4.2 – Análise sobre o Estudo de Caso e o Processo da Área de Testes

O presente estudo de caso atingiu o objetivo proposto de analisar a atuação da área de testes em uma grande empresa de Telecom após o início da transformação ágil e foi possível identificar que melhorias importantes ainda precisam ser feitas para que a área de Gestão de Testes possa se enquadrar melhor nas práticas do *SAFe* e se tornar uma área com mindset ágil.

Abaixo uma lista de problemas identificados e que precisam ser sanados para que essa transformação seja efetiva na área de Gestão de Testes.

- Time de Testes trabalhando de forma híbrida, ou seja, projetos atuando em modo Waterfall e Ágil;
- Squad de Testes não atende todos os sistemas que fazem parte do escopo da área, o que gera muitas das vezes a necessidade de atuar no modo Waterfall porque apenas um sistema não faz parte do escopo da Squad;
- Perfil dos recursos alocados na Squad de testes não são multi skill, ou seja, não atua em sistemas diferentes. Essa limitação é uma premissa contratual do fornecedor.
- Burocracia para iniciar o planejamento dos testes somente após aprovação financeira da uma proposta de testes;
- Divisão de atividades dentro da área como se fossem equipes diferentes;

- Falta de treinamento para os colaboradores para conhecimento do *SAFe*, Kanban, Scrum, ferramentas de apoio aos testes;
- Squads de Desenvolvimento só permite ser composta por sistemas de um mesmo fornecedor, se houver impacto em sistemas de fornecedor de desenvolvimento diferente, a Iniciativa/Projeto não é atendido pela Squad;
- Falta de engajamento das lideranças na mudança do processo;

#### **4.2.1 – Proposta de Ações de Melhorias na Área de Testes**

Como objetivo do trabalho foi um Estudo de Caso, aqui neste capítulo serão apresentadas as propostas para melhorias na Área de Testes.

Diante do exposto acima e do estudo realizado será feita uma proposta para que o processo da Área de Testes seja ajustado.

Certamente não será um trabalho rápido e fácil, pois tem muitas mudanças culturais que precisam ser superadas. E além de tudo o modelo de trabalho da companhia, de ter seu processo de desenvolvimento e testes todo terceirizado, com isso os contratos dos fornecedores deverão ser revisados para melhor aderência a nova forma de trabalho.

A atuação do time de testes deverá ser via time ágil, seguindo os conceitos de contratação de time multidisciplinar.

As *Squads* deverão ser compostas por desenvolvedores, equipe de operação, equipe de segurança da informação e gerencia de qualidade com todas as suas áreas envolvidas (Testes, Release, Gestão de Ambiente Não Produtivo e Garantia da Qualidade), isso irá gerar melhor sinergia. Os testes realizados nessas *Squads* serão os Testes Unitários e Testes de Sistemas. Para os Testes Integrados e UAT deverá ser criada uma *Squad* de Testes multi skill que deverá atuar em paralelo as *Squads* de Desenvolvimento e seu *backlog* controlado a partir da utilização do *Kanban*.

Além dos Testes Integrados e Homologação (UAT), os Testes de Sistemas também deverão fazer parte do escopo de atuação da área de testes, isso irá gerar sinergia nos planos de testes e maior controle de qualidade, além disso o plano de testes deverá ter a visão por produtos, e não mais uma visão por sistemas como é planejado atualmente.

A atualização do processo deverá considerar a realização dos testes contínuos por produto e com isso deverá ser constituído por Testes de Sistemas, Testes Integrados (integração

entre sistemas), Homologação, Testes de Performance, Testes de Segurança (quando aplicável) e Testes de Regressão.

Como a definição de uma *Squad* multidisciplinar e com colaboradores multi skill, o documento de Estratégia de Testes, que é utilizado como insumo para a contratação da Fábrica de Testes, poderá ser descontinuado e deverá ser escritas estórias de testes, baseadas nos critérios de aceitação das estórias de usuário.

O processo de automação de casos de testes deverá ser mais efetivo, com isso quando um plano de testes estiver sendo desenhado deverá considerar se já tem Caso de Teste automatizado, se tiver deve ser reutilizado, se não tiver deve ser imediatamente direcionado para o time de automação trabalhar nesses casos de testes para serem reutilizados futuramente de forma automatizada.

Deverá ser considerado a criação de um Caderno de Testes, que servirá como um catalogo de Casos de Testes para ser reutilizados em iniciativas futuras. A premissa para o bom funcionamento e reuso deste caderno é que os Casos de Testes sejam automatizados.

#### **4.2.2 Revisão dos Papéis do Time de Qualidade**

Com as melhorias previstas deverá ser feita uma revisão dos Papéis e Responsabilidades do Time de Testes, como:

- Test Lead (Colaborador) – Recurso Cross responsável por acompanhar os testes de produto juntamente com elaboração dos planos de testes, aderência do desenvolvimento aos critérios de Qualidade, capitanear os insumos para a automação dos testes e garantir que o desenvolvimento realizado esteja aderente a esteira de DevOps. Participa das PI Planning e organiza o Backlog do time de testes;
- Analista de Qualidade – Planejamento de Testes (Fornecedor – Fábrica de Testes) – Responsável por elaborar o Plano de Testes do Produto baseado nas Estórias de Testes. Participa junto ao PO, participando das reuniões de Planning e Refinamento;
- Tester (Fornecedor – Fábrica de Testes) – Responsável por executar os testes do Produto baseado no Plano de Teste aprovado e registrar os defeitos encontrados durante a execução dos testes. Esse recurso é membro da Squad de Testes.
- Analista de Automação (Colaborador) – Responsável pela gestão e acompanhamento da automação dos cenários de testes (CTs) planejados.

O objetivo dessa mudança é inclusive para ajustar os papéis do time de testes, pois é importante que os membros da equipe trabalhem de forma ágil e colaborativa, portanto o Test Lead poderá atuar em qualquer atividade que estiver no backlog no time e ter autonomia para assumir as atividades sem precisar que alguém direcione o que cada um tem que fazer.

### 4.2.3 – Macro Fluxo do Processo de Gestão de Testes

A figura 4.2.3.1 apresenta o macro fluxo proposto para o novo processo da área de testes. Essas atividades podem acontecer paralelas para projetos diferentes, de acordo com o *Kanban* da Squad de Testes.

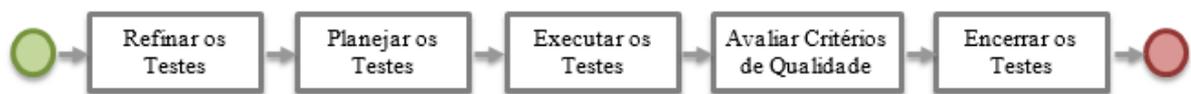


Figura 4.2.3.1 – Macro Fluxo do Processo de Testes

Fonte: Autor

O processo de testes será iniciado quando as USs estiverem refinadas e priorizadas no backlog das Squads de desenvolvimento.

**Refinar os Testes** – Após o resultado da PI, tendo as USs priorizadas e detalhadas o Analista de Qualidade – Planejamento de Testes inicia a escrita das Estórias de Testes baseadas nas Estórias de Usuário, quando as USs estiverem refinadas as Estórias de Testes são finalizadas e enviadas para aprovação;

**Planejar os Testes** – Este subprocesso engloba o planejamento dos testes de Sistemas, Testes Integrados, TRG e Automação e todo planejamento dos Casos de Testes ocorrerá em ferramenta de apoio aos testes.

**Executar os Testes** – Após a finalização do planejamento e o código disponível é iniciada a execução dos testes. Durante a execução são geradas as evidências dos testes e armazenadas na ferramenta de apoio aos testes. Também nesta ferramenta serão registrados os defeitos encontrados durante os testes e tratados pelos respectivos responsáveis.

**Avaliar Critérios de Qualidade** – Após a finalização da execução dos testes as evidências dos testes geradas durante a execução, são validadas e aprovadas pelo POs. Após essa etapa e antes de implantação em produção, serão realizados os testes de regressão geral;

**Encerrar os Testes** – é a última etapa do processo, onde irá acontecer a cerimônia de retrospectiva da Sprint, onde será analisado o que funcionou, o que não funcionou e o que pode melhorar para a próxima entrega.

O objetivo é que a *Squad* de Testes Integrado trabalhe de forma paralela as *Squads* de Desenvolvimento, no entanto o time de testes já fará parte da *Squad* de Desenvolvimento, atuando no planejamento dos testes e na execução do Teste de Sistemas.

O planejamento dos testes será entregue ao time da *Squad* de Testes, que irá executar os Testes Integrados, teste de performance e testes de regressão, conforme o escopo necessário baseado no seu *backlog* controlado pelo *Kanban*.

#### **4.2.4 – Outras Iniciativas para o Sucesso na Implantação do Processo**

Além da reformulação da atuação do time de testes nas *Squads*, também deverão ser realizadas outras ações importantes para que seja possível engajar os colaboradores nas mudanças. Quanto mais engajada e produtiva a equipe estiver, maiores serão os resultados que a empresa como um todo irá obter.

Não adianta mudar os processos e simplesmente exigir que todos atuem no novo formato, sem que os colaboradores sejam devidamente treinados na metodologia, nas ferramentas propostas, e que sejam ouvidos.

Engajamento deve ser incorporado na cultura da empresa através de ações planejadas e propostas pelo RH e gestores, como foco na valorização, motivação e retenção de talentos. Embora o envolvimento do RH demonstre ser mais abrangente, se o gestor da área de testes, atuar em conjunto com RH e investir em treinamentos para o seu time, certamente irá gerar motivação para as mudanças propostas.

Manter um diálogo aberto e implementar ciclos de *feedbacks* são fundamentais para que a cultura de engajamento atinja seu objetivo e com isso manter alinhadas as expectativas para o crescimento da empresa e dos colaboradores.

Mesmo que não seja encontrada uma fórmula padrão para engajar os colaboradores, a implementação de algumas práticas tende a gerar bons resultados, casos sejam utilizadas de forma constante, como por exemplo:

- **Inovação:** Profissionais da área de TI são entusiastas por natureza e gostam de aprender sobre novas tecnologias, processos e tendências e isso não é diferente para profissionais que se desenvolvem na área de Teste de Software. Portanto, investir em treinamentos direcionados para o perfil do colaborador é um passo importante;
- **Gestão de Pessoas:** Uma equipe engajada precisa passar por um processo de gestão de pessoas bem estruturada. Quando um gestor entende o perfil dos seus

colaboradores, suas necessidades individuais e as necessidades coletivas da equipe, ele consegue promover ações para desenvolver o bem-estar individual e coletivos, e isso certamente fará com que os colaboradores se sintam como parte importante para o crescimento da empresa;

- Incentivar a autonomia: A autonomia promove o senso de responsabilidade e decisão do colaborador. Para tornar uma equipe produtiva, investir em modelos flexíveis de trabalho e ter o suporte das lideranças, favorece para que os colaboradores coloquem em prática suas habilidades;
- Comunicação: Uma das formas mais eficazes de manter uma equipe engajada é ter uma comunicação aberta. Promover o diálogo e desenvolver o relacionamento entre as equipes e as lideranças é fundamental para o sucesso de qualquer mudança;
- Feedback: Criar um ciclo de feedbacks para que seja possível orientar o colaborador e ter um diálogo franco sobre o desempenho é importante para manter o time engajado. Portanto deve ser um feedback aberto, onde a liderança também se coloque a disposição para ouvir o que precisa melhorar. Assim, juntos, colaborador e gestor poderão alinhar as expectativas da empresa e das pessoas e com isso obter melhores resultados;
- Construção da Equipe: Construir uma equipe perfeita pode ser utopia, porém se tiver boa vontade em fazer diferente, alinhando as expectativas, organizando as ideias, criando incentivo para retenção de talentos e aceitando sugestões poderá ser possível engajar os times nas mudanças necessárias.

Sair do lugar comum e implementar um novo modelo de trabalho onde o foco no cliente e a entrega rápida de valor seja um objetivo comum de todos os colaboradores e fornecedores da companhia, certamente será um diferencial para que a empresa se mantenha em destaque no mercado e a frente da concorrência.

# Capítulo 5

## Conclusão e Trabalhos Futuros

### 5.1 – Conclusão

Com o estudo que foi realizado neste trabalho sobre os métodos ágeis, foi possível entender que não é necessário escolher apenas um único tipo de framework para passar a utilizar o ágil nas empresas. A junção de melhores práticas de cada um dos métodos poderá possibilitar a utilização do ágil de forma eficaz.

No estudo de caso foi possível identificar que essa junção foi feita, pois a empresa passou a trabalhar com portfólio de produtos que é orientado pelo *SAFe* e foram definidas cerimônias e papéis que são sugeridos no Scrum, portanto ainda é necessário incluir outras competências, como por exemplo o uso de cadeia puxada utilizando o Kanban e seu quadro de tarefas que demonstra de forma clara e visual o andamento das atividades.

Realizar um trabalho junto com os colaboradores, preparando-os para as mudanças e no entendimento das práticas ágeis faz com que todos se sintam envolvidos e preparados para a utilização do Ágil.

O uso de frameworks ágeis requer mudança de cultura e engajamento de todos os envolvidos, desde o mais alto nível executivo até os funcionários “chão de fábrica” e seus fornecedores. Empresas, como a que foi origem deste estudo de casos, que trabalham fortemente com serviços terceirizados, com contratos muito burocráticos podem ter mais dificuldade para realizar mudanças, mas não é impossível. Neste estudo de caso foi possível identificar que etapas importantes no processo de desenvolvimento de software, como é o caso de Testes, precisam de um olhar mais cuidadoso para que a transformação ágil seja realizada com sucesso, por isso a importância em rever o processo da área de Gestão de Testes e a atuação de seus fornecedores.

Não basta apenas realizar mudanças no processo, é necessário que essas mudanças sejam medidas e que demonstrem para os patrocinadores que o investimento para mudar o que já funciona irá trazer benefícios para a companhia. No próximo tópico de Trabalhos Futuros são sugeridos indicadores que poderão ser implementados e dar continuidade a este trabalho.

## 5.2 – Trabalhos Futuros

Este trabalho fez uma abordagem para melhorias de uma área específica que é a Área de Gestão de Testes e como essa etapa importante dentro do ciclo de vida de desenvolvimento poderá atuar na transformação ágil que está sendo realizada na empresa.

Para trabalhos futuros, poderá ser feita uma abordagem mais ampla considerando todas as demais áreas da Gerência de Qualidade, como por exemplo a área de Gestão de Release e a Área de Garantia da Qualidade. Essas áreas poderão aprimorar seus processos utilizando as competências do *SAFe*, como por exemplo Entrega de Solução Empresarial, que orienta o uso do DevOps por meio do *Continuous Delivery Pipeline* que permite fornecer entregas mais rápidas e contínuas de valor. E é importante investir em com ciclos mais curtos de implantação.

Certamente a Gerência de Qualidade possui grandes oportunidades de melhorias que poderão contribuir para que a empresa mantenha seu engajamento para uma verdadeira transformação ágil.

Além de aprimorar a mudança do processo para as demais áreas da gerência de qualidade é de suma importância que sejam estabelecidos indicadores para apurar a qualidade dos serviços realizados e os ganhos que a empresa obtém com a evolução dos seus processos.

Como sugestão seguem alguns indicadores que podem ser implementados futuramente:

- Tempo de Correção de Defeitos – medir a qualidade e tempo de atendimento dos fornecedores de desenvolvimento quando forem abertos defeitos durante os testes;
- Indicador de Velocidade - métrica para monitorar a velocidade de entregas das estórias das Sprints por Squad;
- Densidade de Erro por Sistema - medir a qualidade do código entregue pelos fornecedores de desenvolvimento;
- Taxa de Defeitos em UAT - Avalia a qualidade da atuação da área de testes e como foi a qualidade dos testes realizados em etapas anteriores, como o Teste de Sistemas e Testes Integrados;

# Referências Bibliográficas

[1] PIKIILA, Jéssica. O que é o *SAFe*: Aprenda sobre o *SAFe* e seus princípios e entenda como ele se diferencia de outras estruturas ágeis. c2021. Disponível em: <https://www.atlassian.com/br/agile/agile-at-scale/what-is-SAFE>. Acesso em: 30/05/2021

[2] Disponível em: <https://www.scaledagileframework.com/posters/> c2010-2021. Acesso em: 25/05/2021

[3] Disponível em: <https://www.scaledAgileframework.com/enterprise-solution-delivery/> c2010-2021. Acesso em: 27/05/2021

[4] Disponível em: <https://www.scaledAgileframework.com/Agile-product-delivery/> c2010-2021. Acesso em: 30/05/2021

[5] Disponível em: <https://www.scaledAgileframework.com/team-and-technical-agility/> c2010-2021. Acesso em: 30/05/2021

[6] Disponível em: <https://www.scaledAgileframework.com/Lean-Agile-leadership/> c2010-2021. Acesso em: 03/06/2021

[7] BONILHA, Diego. OS 4 VALORES DO SCALED AGILE FRAMEWORK (*SAFe*). Disponível em: <https://blog.adaptworks.com.br/2021/04/os-4-valores-do-scaled-Agile-framework-SAFE/> c2009. Acesso em: 02/06/2021

[8] BARBOSA, Anderson Baum. *SAFe* - Princípios e Valores Lean-Agile. c2021. Disponível em: <https://www.linkedin.com/pulse/SAFe-princ%C3%ADpios-e-valores-lean-agile-anderson-barbosa-pmp/?originalSubdomain=pt>. Acesso em: 02/06/2021.

[9] Disponível em: <https://www.scaledAgileframework.com/SAFe-Lean-Agile-principles/> c2010-2021. Acesso em: 03/06/2021

- [10] Disponível em: <https://www.scaledagileframework.com/take-an-economic-view/c2010-2021>. Acesso em 03/06/2021
- [11] Disponível em: <https://www.scaledagileframework.com/build-incrementally-with-fast-integrated-learning-cycles/c2010-2021>. Acesso em 03/06/2021
- [12] Disponível em: <https://www.scaledAgileframework.com/continuous-learning-culture/c2010-2021>. Acesso em: 14/06/2021
- [13] Disponível em: <https://www.scaledAgileframework.com/organizational-agility/c2010-2021>. Acesso em: 16/06/2021
- [14] Disponível em: <https://www.scaledAgileframework.com/Lean-portfolio-management/c2010-2021>. Acesso em: 17/06/2021
- [15] Disponível em: <https://www.significados.com.br/Kanban/>. C2011-2021. Atualização Dez/2019. Acesso em: 19/06/2021
- [16] KNIBERG, Henrik; SKARIN, Mattias. Kanban e Scrum obtendo o melhor de ambos. C2021. Disponível em: [https://www.academia.edu/26269944/Kanban E Scrum - InfoQBrasil](https://www.academia.edu/26269944/Kanban_E_Scrum_-_InfoQBrasil). Acesso em: 09/07/2021
- [17] PEREIRA, Daniel. A importância da visualização no gerenciamento de projetos. C2021. Disponível em: <https://blog.myScrumhalf.com/a-importancia-da-visualizacao-no-gerenciamento-de-projetos>. Acesso em: 22/06/2021
- [18] Disponível em: <https://Kanbanize.com/pt/recursos-Kanban/primeiros-passos/o-que-e-Kanban>. C2021. Acesso em: 20/06/2021
- [19] SCHWABER, Ken; SUTHERLAND, Jeff. **Guia do Scrum: Um guia definitivo para o Scrum**: As regras do jogo. Julho 2013.
- [20] RIBEIRO, Wagner. **UFRJ - MGPS - Gerenciamento Ágil em Projetos**. 2020, p. 77 e 90

[21] Disponível em

**[https://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o\\_extrema](https://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o_extrema)**. 2021. Acesso em: 17/07/2021

[22] HIGOR. Práticas em XP: Extreme Programming. 2013. Disponível em:

**<https://www.devmedia.com.br/praticas-em-xp-extreme-programming/29330>**. Acesso em: 17/07/21

[23] Disponível em **[https://pt.wikipedia.org/wiki/Teste\\_de\\_software#cite\\_note-1](https://pt.wikipedia.org/wiki/Teste_de_software#cite_note-1)**. 2021.

Acesso em: 19/07/21

[24] Disponível em: **[https://pt.wikipedia.org/wiki/ISO/IEC\\_9126#Funcionalidade](https://pt.wikipedia.org/wiki/ISO/IEC_9126#Funcionalidade)**. 2020.

Acesso em 19/07/21