



PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PROCESSOS  
MESTRADO PROFISSIONAL

**ADAPTAÇÃO DA METODOLOGIA DA INTEGRAÇÃO CONTÍNUA NA  
ESTRATÉGIA DE DESENVOLVIMENTO DE PRODUTOS DE SOFTWARE  
EM EMPRESAS DE ECONOMIA MISTA DE TECNOLOGIA – ESTUDO DE  
CASO**

**Renato Borges de Souza**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia de Processos – Mestrado Profissional, PPGEP/ITEC, da Universidade Federal do Pará, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Processos.

Orientador: Manoel Henrique Reis Nascimento

Belém

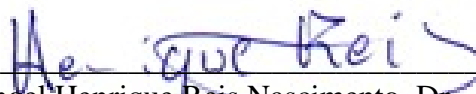
Setembro de 2021

**ADAPTAÇÃO DA METODOLOGIA DA INTEGRAÇÃO CONTÍNUA NA  
ESTRATÉGIA DE DESENVOLVIMENTO DE PRODUTOS DE SOFTWARE  
EM EMPRESAS DE ECONOMIA MISTA DE TECNOLOGIA – ESTUDO DE  
CASO**

Renato Borges de Souza

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PROCESSOS – MESTRADO PROFISSIONAL (PPGEP/ITEC) DA UNIVERSIDADE FEDERAL DO PARÁ COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA DE PROCESSOS.

Examinada por:



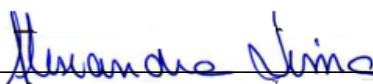
---

Prof. Manoel Henrique Reis Nascimento, Dr.  
(PPGEP/ITEC/UFPA-Orientador)



---

Prof. Jandecy Cabral Leite, Dr.  
(PPGEP/ITEC/UFPA-Membro)



---

Prof. Alexandra Amaro de Lima, Dra.  
(UNIP-Membro)

BELÉM, PA - BRASIL  
SETEMBRO DE 2021

**Dados Internacionais de Catalogação na Publicação (CIP)  
Sistema de Bibliotecas da UFPA**

---

Souza, Renato Borges de, 1982 -  
Adaptação da metodologia da integração contínua na  
estratégia de desenvolvimento de produtos de software em  
empresas de economia mista de tecnologia - Estudo de caso /  
Renato Borges de Souza - 2021.

Orientador: Manoel Henrique Reis Nascimento

Dissertação (Mestrado Profissional) - Universidade  
Federal do Pará, Instituto de Tecnologia, Programa de Pós-  
Graduação em Engenharia de Processos, Belém, 2021.

1. Integração contínua. 2. Processo de desenvolvimento  
de produtos. 3. Processo de negócio. 4. Agilidade. 5. Devops.  
I. Título.

CDD 620

---

*Dedico este trabalho primeiramente ao meu pai (in memoriam), grande exemplo e líder, a mãe e esposa, pelo incentivo de sempre, ao meu orientador por toda dedicação na entrega de resultados.*

## **AGRADECIMENTOS**

Primeiramente gostaria de agradecer a Deus por ter me mantido no caminho de conclusão da dissertação, mesmo com tantos desafios.

Agradeço ao meu orientador Dr. Manoel Henrique Reis Nascimento por aceitar conduzir o meu trabalho de dissertação, sempre disponível e com enorme atenção aos detalhes.

Agradecimento especial aos meus pais Roberto Gonçalves de Souza, que encontra-se ao lado de Deus, e Maria Aurora Borges de Souza, que, mesmo nas condições mais difíceis, sempre estiveram ao meu lado me apoiando ao longo de toda a minha trajetória.

À minha esposa Michely Souza Borges pela compreensão e paciência demonstrada durante o período do projeto. Aos meus filhos Maria Isis Souza Borges e Ryan Souza Borges, por, mesmo sem saber, incentivar a seguir em frente na conclusão da dissertação.

Aos meus amigos Fábio Reginaldo e Thulio Queiroz, pelos momentos de direcionamento técnico, trazendo sempre uma visão de mercado para dentro da dissertação.

*“Não deixe o barulho da opinião dos outros abafar sua voz interior. E mais importante, tenha a coragem de seguir seu coração e sua intuição. Eles de alguma forma já sabem o que você realmente quer se tornar. Tudo o mais é secundário”.*

(Steve Jobs)

Resumo da Dissertação apresentada ao PPGEP/UFPA como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia de Processos (M. Eng.)

**ADAPTAÇÃO DA METODOLOGIA DA INTEGRAÇÃO CONTÍNUA NA  
ESTRATÉGIA DE DESENVOLVIMENTO DE PRODUTOS DE SOFTWARE  
EM EMPRESAS DE ECONOMIA MISTA DE TECNOLOGIA – ESTUDO DE  
CASO**

**Renato Borges de Souza**

Setembro/2021

Orientador: Manoel Henrique Reis Nascimento

Área de Concentração: Engenharia de Processos

A metodologia de integração contínua é um amplo campo de atuação que lida com a codificação e publicação de códigos de sistemas, que visa ajudar as organizações a acelerar seu desenvolvimento e entrega de recursos de software, sem comprometer a qualidade. Esta abordagem tornou-se extremamente importante para dar celeridade nas entregas de produtos. O foco principal desse artigo é promover a adaptação dessa metodologia através do alinhamento com a estratégia de desenvolvimento de produtos de software, de forma a melhorar os resultados organizacionais, através da redução de desperdício, com o conhecimento do processo ponta a ponta. Um estudo de caso foi proposto para ilustrar a execução desse processo. Onde, primeiramente, foi feito o mapeamento e análise do processo de desenvolvimento de sistemas e de gestão de portfólio vigente na empresa, identificando sua problemática, e em seguida foi proposto um desenho de processo futuro, promovendo a integração entre as partes. O referido processo ajudará a empresa a entregar produtos alinhados a estratégia organizacional.

Abstract of Dissertation presented to PPGE/UFPA as a partial fulfillment of the requirements for the degree of Master in Process Engineering (M. Eng.)

**ADAPTATION OF THE CONTINUOUS INTEGRATION METHODOLOGY IN  
THE SOFTWARE PRODUCT DEVELOPMENT STRATEGY IN MIXED  
ECONOMY TECHNOLOGY COMPANIES - CASE STUDY**

**Renato Borges de Souza**

November/2021

Advisor: Manoel Henrique Reis Nascimento

Research Area: Process Engineering

The continuous integration methodology is a broad field that deals with the availability and publication of system codes, which aims to help associations to accelerate their development and delivery of software resources, without compromising quality. This approach becomes extremely important for speed in product deliveries. The main focus of this article is to promote the adaptation of this methodology by aligning it with the software product development strategy, in order to improve organizational results, through waste reduction, with knowledge of the end-to-end process. A case study was proposed to illustrate an execution of this process. Where, first, the mapping and analysis of the system development process and current portfolio management in the company was carried out, identifying its problems, and then a future process design was proposed, promoting the integration between the parts. This process will help the company to deliver products that fit the organizational strategy.



## SUMÁRIO

<b>CAPÍTULO 1 - INTRODUÇÃO.....</b>	<b>1</b>
1.1 - MOTIVAÇÃO.....	1
1.2 - OBJETIVOS.....	3
<b>1.2.1 - Objetivo geral.....</b>	<b>3</b>
<b>1.2.2 - Objetivos específicos.....</b>	<b>4</b>
1.3 - CONTRIBUIÇÕES DA DISSERTAÇÃO.....	4
1.4 - ORGANIZAÇÃO DO TRABALHO.....	4
<b>CAPÍTULO 2 - REVISÃO DA LITERATURA.....</b>	<b>6</b>
2.1 - GESTÃO POR PROCESSOS.....	6
<b>2.1.1 - Mapeamento de processos de negócios e a notação BPMN.....</b>	<b>7</b>
2.2 - PROCESSO DE DESENVOLVIMENTO DE PRODUTOS.....	9
2.3 - PROCESSO DE DESENVOLVIMENTO DE SOFTWARE E AGILIDADE	11
2.4 - INTEGRAÇÃO CONTÍNUA.....	13
<b>CAPÍTULO 3 - MATERIAIS E MÉTODOS.....</b>	<b>17</b>
3.1 - MATERIAS.....	17
3.2 - CICLO DA PESQUISA.....	17
3.3 - FERRAMENTAS UTILIZADAS.....	19
3.4 - SOFTWARES UTILIZADOS.....	19
<b>CAPÍTULO 4 - RESULTADOS E DISCUSSÃO.....</b>	<b>20</b>
4.1 - CARACTERIZAÇÃO DA EMPRESA.....	20
<b>4.1.1 - Descrição do processo de gerenciamento de portfólio de produtos.....</b>	<b>20</b>
<b>4.1.2 - Descrição do processo de desenvolvimento de sistemas.....</b>	<b>22</b>
4.2 - CARACTERIZAÇÃO DO PROBLEMA.....	24
4.3 - PROPOSTA DE PROCESSO DE DESENVOLVIMENTO DE PRODUTO DE SOFTWARE.....	27
<b>4.3.1 - Processo atual.....</b>	<b>27</b>
<b>4.3.2 - Processo proposto de desenvolvimento de produto de software.....</b>	<b>27</b>
4.3.2.1 - Matriz do processo por área.....	28
4.3.2.2 - Mapa geral do processo.....	28
4.3.2.3 - Indicadores.....	29
4.3.2.4 - Descrição das atividades.....	30

<b>4.3.3 - Modificações propostas através do novo processo.....</b>	<b>32</b>
<b>CAPÍTULO 5 - CONCLUSÕES E SUGESTÕES.....</b>	<b>33</b>
5.1 - CONCLUSÕES.....	33
5.2 - SUGESTÕES.....	34
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>35</b>
<b>ANEXO I - PROCESSO DESENVOLVIMENTO DE PRODUTO DE SOFTWARE.....</b>	<b>37</b>

## LISTA DE FIGURAS

Figura 2.1	Processo adquirir bens e serviços.....	8
Figura 2.2	Componentes de um sistema de integração contínua.....	15
Figura 3.1	Descrição das fases de pesquisa.....	19
Figura 4.1	Processo criar produto.....	21
Figura 4.2	Processo desenvolvimento e manutenção de sistemas.....	22
Figura 4.3	Fluxo de trabalho Git.....	24
Figura 4.4	Processo desenvolvimento de produto de software.....	28

## LISTA DE TABELAS

Tabela 3.1	Modelo de tabela de coleta de dados.....	18
Tabela 4.1	Total de chamados por tipo/ano.....	25
Tabela 4.2	Mapa geral do processo.....	28
Tabela 4.3	Indicadores do processo.....	29

## NOMENCLATURA

ABNT	ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS
BPM	<i>BUSINESS PROCESS MANAGEMENT</i>
BPMN	<i>BUSINESS PROCESS MODEL AND NOTATION</i>
CI	<i>CONTINUOUS INTEGRATION</i>
CD	<i>CONTINUOUS DEPLOYMENT</i>
CDE	<i>CONTINUOUS DELIVERY</i>
DEVOPS	<i>DEVELOPMENT AND OPERATIONS</i>
OMG	<i>OBJECT MANAGEMENT GROUP</i>
PDP	PROCESSO DE DESENVOLVIMENTO DE PRODUTOS
SGQ	SISTEMA DE GESTÃO DA QUALIDADE

# CAPÍTULO 1

## INTRODUÇÃO

### 1.1 - MOTIVAÇÃO

Alta concorrência, escopo de produto variável, mudanças constantes, tem feito as empresas de software buscarem maneiras cada vez mais ágeis de lançarem seus produtos no mercado e obterem melhores resultados. Segundo HUMBLE *et al.* (2016), em uma época em que vantagem competitiva requer tempo rápido para o mercado e implacável experimentação, as organizações incapazes de replicar esses resultados são destinadas a perder no mercado para concorrentes mais ágeis e poderiam potencialmente sair inteiramente do negócio. Com o avanço da tecnologia das linguagens de programação, infraestrutura de servidores de aplicação, computação em nuvem, processos de desenvolvimento de sistemas, integração de dados, Internet, gestão de projetos, as empresas de software têm se deparado com o seguinte desafio: como entregar mais, com valor e mais rápido.

O negócio da empresa é formado por um grupo de setores que devem trabalhar em conjunto através da realização de processos de negócios para atingir o resultado almejado, onde os mesmos devem ser conhecidos, gerenciados e aprimorados continuamente para manter a empresa competitiva. O processo de negócio é um trabalho que entrega valor para os clientes ou apoia/gerencia outros processos, sendo o processo uma agregação de atividades e comportamentos executados por humanos ou por máquinas para alcançar um ou mais resultados (BENEDICT *et al.*, 2013). Para ROZENFELD *et al.* (2006), o processo de desenvolvimento de produto, constitui-se em um dos processos-chave de qualquer empresa que se proponha a competir por meio da criação de produtos próprios e da busca por liderança tecnológica. O termo “produto” é um elemento que faz parte de uma proposta da empresa para aproveitar alguma oportunidade de mercado, para satisfazer um desejo ou uma necessidade (MATTAR *et al.*, 2014). Nesse sentido, processos de desenvolvimento de produtos, quando usados de forma consistente, são um diferencial para as empresas que os utilizam. Principalmente no cenário de agilidade e extrema competição, com necessidade de lançamento contínuo de produtos, é imperativo que a empresa tenha processos definidos e integrados de entrega de produtos. Isso envolve ciclos desde a análise da necessidade dos clientes, definição de

requisitos, programação de códigos para sistemas e aplicativos, testes de segurança, testes funcionais, publicação em produção, avaliação de infraestrutura, acompanhamento do uso do cliente, marketing, medição de resultados financeiros até a retirada do produto do mercado.

Olhando mais particularmente em empresas de software, os processos de entrega de produtos estão partindo de uma visão mais prescritiva, para uma mais adaptativa. Segundo BRAGA (2015), as metodologias de desenvolvimento de software ditas como prescritiva, atenderam de forma única e adequada às necessidades de negócio relativas àquela época, onde, na maioria dos casos, os softwares não exigiam muitas mudanças nos requisitos, o que, inicialmente, eram suportados por esses modelos. Essa característica mudou e a mudança tornou-se rotineira nesse cenário. Para KRUCHTEN (2013), agilidade é descrita como a capacidade de uma organização reagir às mudanças, em seu ambiente, mais rapidamente do que à taxa dessas mudanças. Um dos princípios do manifesto ágil (2001), é aceitar as mudanças de requisitos, mesmo que no fim do desenvolvimento. Esse princípio pode servir como base para os outros 11 princípios, entre os quais destacamos o que informa que a maior prioridade é satisfazer o cliente, através de entrega adiantada e contínua de software de valor.

Diante dessa necessidade, as empresas têm dado mais atenção aos processos de desenvolvimento de produtos, aproximando as áreas produtivas em prol da agilidade, além do uso de ferramentas tecnológicas como virtualização, computação em nuvem e infraestrutura como código, visando o desenvolvimento e entrega de software com qualidade e de forma acelerada. Segundo SHAHIN *et al.* (2017), Integração Contínua (CI – *Continuous Integration*), Entrega Contínua (CDE – *Continuous DELivery*) e Implantação Contínua (CD- *Continuous Deployment*), chamados de práticas contínuas de desenvolvimento de software, são algumas das práticas que visam ajudar as organizações a acelerar seu desenvolvimento e entrega de recursos de software sem comprometer a qualidade. Em outro estudo, desenvolvido por STAHL (2016), foi identificado que capacidades de integração e entregas contínuas, foram consideradas cada vez mais importantes para as indústrias, com grande quantidade de dinheiro e recursos sendo investidos na busca dessa integração, visto que, a maioria das vezes são soluções sob medida para cada empresa, sendo que, apesar da importância e do investimento realizado, o problema de construir produtos de integração eficazes e eficientes não são abordados necessariamente de forma sistêmica. Nesse sentido, planejar, mapear, analisar, desenhar, monitorar e controlar o processo de entrega de produtos, desde a decisão por iniciá-lo até

seu uso por parte do cliente, envolve não apenas uma visão técnica, mas sim uma visão de negócios. O referido processo torna-se um importante trabalho na busca por melhores resultados na empresa, pois produtos entregues pela equipe técnica, sem o devido relacionamento à necessidade dos negócios, não servirão de nada, sendo muitas das vezes desperdício. Identificar, por exemplo, os atores envolvidos no processo, o tempo total de trabalho, tempo por atividade, quantidade de *deploys*, quantidade de falhas na entrada do sistema em produção, entre outras métricas envolvidas no processo, permitirão uma melhor análise do cenário atual da empresa e onde ela deverá ajustar para atingir suas metas.

Dentro desse cenário, as empresas vêm enfrentando problemas na integração de processos de trabalho relacionados a entrega de produtos, desde a concepção de novos modelos de negócios, desenvolvimento e publicação em ambientes de produção com novas versões de produto de software. Tem-se como hipótese a não existência de um processo de desenvolvimento de produtos, alinhado, de forma transparente, envolvendo todas as partes produtivas da organização. Sendo necessário uma adaptação nos referidos processos, tendo um olhar com foco no cliente, no sentido de analisá-lo para identificar o que é valor percebido, permitindo ter um melhor conhecimento sobre a abrangência do produto, gerir melhor as mudanças em necessidades do cliente, a identificação de pontos de melhoria em operações na infraestrutura, para melhorar o feedback sobre a situação do produto, com uso de ferramentas integradas e colaborativas de desenvolvimento, reduzindo os ciclos de produtos entregues. A utilização de metodologias e ferramentas que aproximem o cliente da equipe técnica de desenvolvimento do produto, e este da equipe de infraestrutura, de forma integrada, será um possível caminho para redução de desperdício, entrega de software mais rápido e com qualidade, proporcionando assim aumento das receitas da empresa, atingindo resultados estratégicos.

## 1.2 - OBJETIVOS

### 1.2.1 - Objetivo geral

Adaptar a metodologia de integração contínua na estratégia do desenvolvimento de produtos de software em empresas de economia mista de tecnologia.



## 1.2.2 - Objetivos específicos

- Descrever o processo macro de integração contínua de software existente da empresa estudo de caso;
- Analisar o modelo de processos presentes na empresa de software utilizada como estudo de caso;
- Adaptar o modelo de entrega de produto de software, com abordagem de integração contínua no processo, realizando a integração com a estratégia.
- Analisar o modelo proposto

## 1.3 - CONTRIBUIÇÕES DA DISSERTAÇÃO

Para a comunidade, este estudo visa apresentar de forma prática, como a integração contínua e o processo de desenvolvimento de produtos são utilizados e como podem contribuir para o sucesso da organização, podendo ser replicado em empresas de tecnologia de governo.

Para implantação e uso efetivo da gestão por produtos, passa por um trabalho de identificação das principais dificuldades empresariais, engajamento e comprometimento dos colaboradores e seus respectivos setores, desta forma, este trabalho contribuirá para elevação da confiança e moral dos envolvidos, criação de um processo com foco em entrega de valor, promovendo assim melhorias no ambiente de trabalho.

## 1.4 - ORGANIZAÇÃO DO TRABALHO

O presente trabalho é composto por cinco capítulos. Este capítulo é o primeiro e discorre sobre a contextualização, motivação, os objetivos, as contribuições da dissertação e a forma de organização do trabalho.

No segundo é apresentado o referencial teórico, onde são relatados os princípios da gestão por processos, processo de desenvolvimento de produto, processos de desenvolvimento de software, agilidade e integração contínua, em seguida são descritas as ferramentas adotadas na empresa estudada. Os procedimentos metodológicos são apresentados no terceiro capítulo detalhando a forma como o trabalho foi conduzido. A discussão dos resultados é realizada no quarto capítulo, apresentando a adaptação da

integração contínua no desenvolvimento de produtos e os resultados obtidos com a aplicação do processo proposto, além dos desafios encontrados ao longo do seu processo de implementação, fornecendo subsídios para o último capítulo. No quinto capítulo são apresentadas as conclusões do trabalho, explanando os principais aspectos abordados, evidências do atendimento aos objetivos propostos e, em função das delimitações deste estudo de caso e da abrangência do tema, são sugeridas pesquisas que possam gerar mais conhecimento na área.

## CAPÍTULO 2

### REVISÃO DA LITERATURA

#### 2.1 - GESTÃO POR PROCESSOS

Estruturas hierárquicas (pirâmides) de poder, que eram divididas em departamentos estanques, cada qual cuidando de sua tarefa, eram características de empresas no início do século XX. Segundo CARVALHO *et al.* (2012), nos últimos anos, as empresas começaram a sofrer perdas em competitividade apesar de estarem bem organizadas conforme modelo funcional. Dentre os fatores que causam essa perda, tem-se a concorrência com empresas internacionais dentro de um mundo globalizado, necessidade de rapidez nas mudanças e a contínua evolução tecnológica, tais como a rede mundial de computadores. Ou seja, não é mais adequado, em um mundo competitivo como o atual, organizar uma empresa em torno de funções e, trabalho em torno de tarefas.

A gestão por processos é, por definição, uma metodologia para avaliação contínua, análise e melhoria do desempenho dos processos que exercem mais impacto na satisfação dos clientes e dos acionistas. Segundo BENEDICT *et al.* (2009), o gerenciamento de processos de negócio, ou BPM - *Business Process Management* - é conceituado como a ação de identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio automatizados ou não, para alcançar os resultados pretendidos, de forma consistente e alinhada com as metas estratégicas de uma organização. A ISO 9000:2000 afirma que, para as organizações funcionarem de forma eficaz, elas devem identificar e gerenciar processos inter-relacionados e interativos. Neste contexto, conforme abordado por MENDOZA *et al.* (2017), processos são tratados como uma coleção inter-relacionada de atividades de trabalho, iniciadas em resposta a um evento, alcançando um resultado específico para um cliente e outros patrocinadores do processo. O mesmo autor informa que, a partir dos modelos de processo de negócio busca-se representar um conjunto de elementos de negócio e suas relações necessárias para atingir um objetivo organizacional. Quando a empresa opta por adotar a disciplina de gestão por processos de negócios, a mesma visa ter uma visão ponta a ponta de como entregar valor ao cliente, estando orientada por processos e não por tarefas, superando as barreiras dos silos setoriais, identificando como estes trabalham em conjunto em detrimento ao seu trabalho estritamente departamental.

### 2.1.1 - Mapeamento de processos de negócio e a notação BPMN

Para representar modelos de processos de negócios, a notação BPMN (*Business Process Model and Notation*) é a de maior destaque. BPMN é um padrão criado pela *Business Process Management Initiative* (BPMI) em 2004, incorporado ao *Object Management Group* (OMG), grupo que estabelece padrões para sistemas de informação, como uma especificação aberta para a modelagem de processos de negócios, com objetivo de fornecer uma notação que pudesse ser facilmente utilizada e compreendida por todos os usuários da organização. Os símbolos utilizados descrevem relacionamentos claramente definidos, tais como fluxo de atividade e ordem de precedência.

Os principais elementos que compõe a notação são:

- Tarefas: que representam um trabalho executado, podendo ser tanto uma tarefa como um subprocesso. É representado por um retângulo com as bordas arredondadas.
- Evento: é algo que acontece durante a execução do processo de negócio, podendo ser do tipo inicial, intermediário ou final. É representado por um pequeno círculo, podendo ele ter borda dupla, mais grossa ou todo preenchido, dependendo de sua finalidade representativa.
- *Gateway*: utilizado para representar as decisões na execução do processo e os possíveis caminhos que ele pode prosseguir. É representado por um losângulo, podendo ser utilizado para dividir ou juntar trajetos possíveis do processo.
- Fluxo de sequência: uma seta utilizado para mostrar a ordem que as atividades e subprocessos devem ser executadas dentro de um processo.
- Piscinas (*pools*): representa a descrição resumida (ou o nome) de um processo, onde os atores do processo irão atuar. É representado por um grande retângulo onde os objetivos da modelagem serão inseridos.
- Raias (*lanes*): são a subdivisão das piscinas e representam os atores (ou setores) pela qual o processo flui durante sua execução. É representado por um retângulo dentro de da piscina.

Conforme apresentado na Figura 2.1, tem-se o exemplo de um processo utilizando a notação BPMN, sendo construído usando a ferramenta de modelagem *Bizagi Modeler*. O processo em questão é o “PROCESSO ADQUIRIR BENS E SERVIÇOS”, onde o início dele é no setor “Dpto. Planejamento”, representado com o evento “Necessidade

Identificada”. Após isso, é realizada a atividade “Planejar” cujo artefato gerado é o “Plano de Compras”. Seguindo a seta do fluxo de seqüência, realiza-se assim a atividade “Comprar”, gerando o “Pedido de Compras”. Em seguida é utilizado um Gateway de junção para receber informações caso, após “Receber” o produto pelo “Dpto. De Compras”, seja identificado alguma anormalidade no produto. Caso esteja tudo bem, é realizado a atividade “Pagar” pelo “Dpto. de Tesouraria” e o processo é finalizado com o evento “Produto adquirido”.

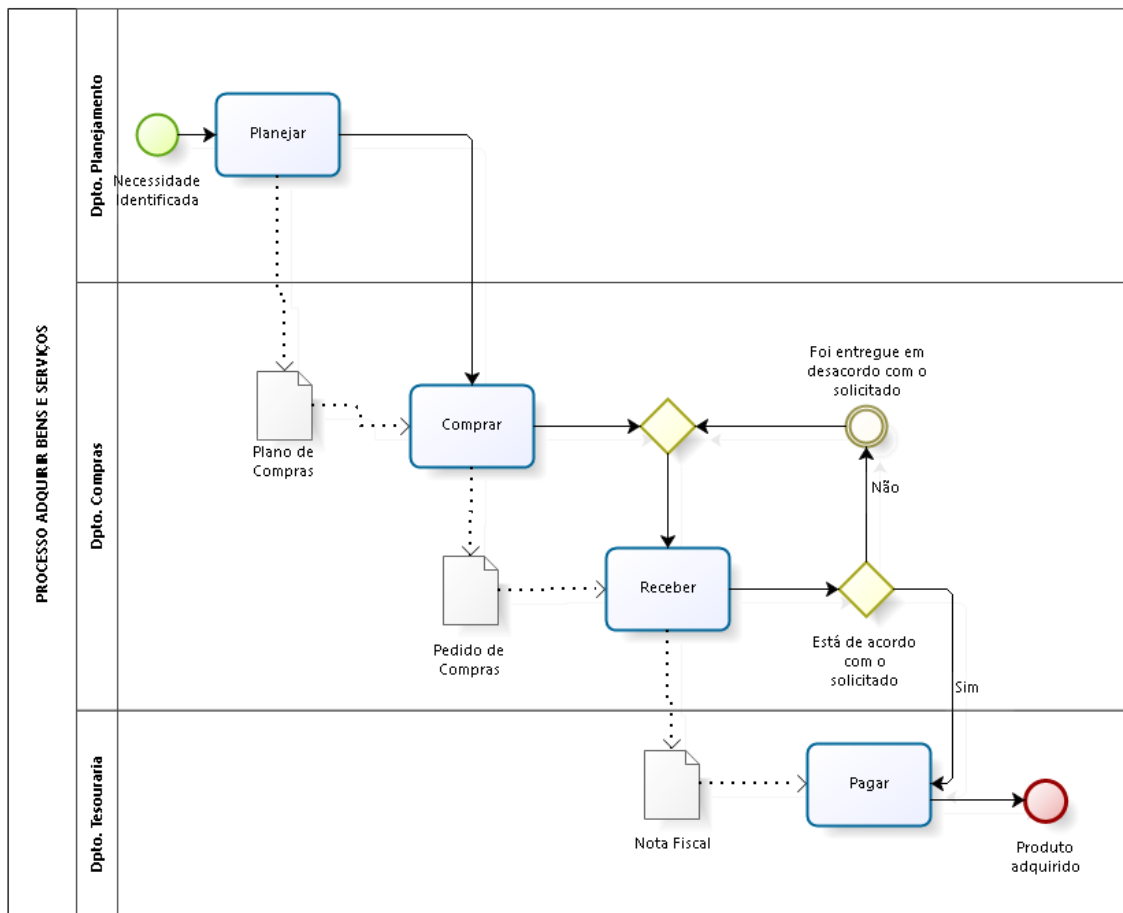


Figura 2.1 - Processo adquirir bens e serviços.

O uso de modelos permite assim representar como acontece a comunicação entre os atores do processo, desde o surgimento de uma necessidade até a entrega do produto final do processo. O mapeamento de processos é uma tarefa muito importante na gestão por processos, pois permite que sejam conhecidas com detalhe e profundidade todas as operações que ocorrem durante a fabricação de um produto ou a produção de um serviço (CARVALHO, 2009). Ela permite descobrir a “fábrica oculta”, entendendo os conceitos do processo e sistema, os seus elementos (fornecedor, entrada, processo, saída, cliente) e

principalmente entender o que é valor para a empresa e o cliente. O mapeamento de processos de negócios utilizando a notação BPMN pode representar tanto como a empresa está funcionando no momento, conhecido como modelo “AS-IS”, tanto como o processo futuro a ser implantando na organização, após análise da situação atual, conhecido como modelo “TO-BE”.

Nesse sentido, para se adequar aos desafios do mercado, quanto mais a empresa conhecer seus processos de negócios e sua ligação com a estratégia, mais assertiva será sua abordagem de melhoria de processos. Como parte do presente trabalho, serão avaliados os processos existentes e mapeados da empresa, entendendo suas fronteiras, onde começam e onde os mesmos são concluídos, referentes ao desenvolvimento de produtos de software.

## 2.2 - PROCESSO DE DESENVOLVIMENTO DE PRODUTOS

Para ROZENFELD *et al.* (2006), desenvolver produtos consiste em um processo (conjunto de atividades) por meio das quais busca-se, a partir das necessidades do mercado e das possibilidades e restrições tecnológicas, e considerando as estratégias competitivas e de produtos da empresa, chegar às especificações de projeto de um produto e de seu processo de produção. Segundo MORGAN (2006), o sistema enxuto de desenvolvimento de produto, caracteriza-se pelo processo prático, sendo as atividades do dia a dia pelas quais a informação flui, os projetos evoluem, os testes são completados, protótipos concluídos, até, enfim, emergir de tudo isso um produto acabado. Conforme exposto por MENDES (2008), a definição do processo de desenvolvimento de produtos (PDP) passou por uma evolução que representou o aumento do seu escopo, indicando uma mudança da visão técnica proporcionada pelo enfoque da engenharia para uma abordagem estratégica e gerencial. Ainda para o autor, o PDP tem como característica ser um processo de negócio crítico para a competitividade das empresas, ser um processo de transformação de informações, o qual inicia-se com a busca de oportunidade de mercado e tecnológicas e se estende por todo o ciclo de vida do produto e é um processo no qual interagem diversos atores organizacionais.

A importância da gestão do processo de desenvolvimento de produto vem do fato dele ser a ligação entre a estratégia e sua operação, permeando por diversas áreas da empresa. Conforme tratado por JUNIOR *et al.* (2012), definir estratégias alinhadas aos processos de negócio da organização e ao mercado, mostra-se, gradativamente, com um

dos processos fundamentais para a geração de valor para a empresa, possibilitando que todos os seus esforços da organização sejam canalizados para atingir metas e objetivos. O mesmo autor afirma ainda que o PDP é estratégico e cada vez mais dependente da tecnologia da informação (TI), uma vez que transita entre os ambientes externo e interno da empresa e, portanto, requer forte alinhamento nas suas ações.

Para SOUMANLAINEN *et al.* (2011), o ambiente de negócios é caracterizado por clientes cada vez mais exigentes, ciclos de vida do produto cada vez mais curtos e tecnologias em rápido desenvolvimento, sendo importante uma visão geral do produto e de seus lançamentos, bem como a existência de processos de desenvolvimento de software bem coordenado e executado para geração de produtos confiáveis e de alta qualidade, dentro do prazo e orçamento. Mas, apesar da sua importância, a sinergia do PDP com os outros setores tem sido difícil de se conquistar na prática por causa da natureza não estruturada desse processo. Por conta disso, o primeiro passo para o gerenciamento eficiente do PDP é torna-lo “visível” para todos os atores envolvidos.

Para TYAGI *et al.* (2015), o qual pesquisou diversos modelos de PDP na literatura, os mesmos podem ser representados por algumas fases principais a depender do autor, podendo conter fases como: geração de conceito, planejamento de produtos, engenharia de produtos e engenharia de processos; geração de conceito, design no nível do sistema, design detalhado, teste e refinamento e aumento da produção, por exemplo. ROZENFELD *et al.* (2006), desenvolveu um modelo de referência composta por três macrofases: pré-desenvolvimento, desenvolvimento e pós-desenvolvimento, onde cada fase faz uma entrega para o prosseguimento de trabalho na fase seguinte. Os processos dentro de cada fase no modelo de referência irão depender do trabalho da empresa, como por exemplo, as atividades de pesquisa de mercado, plano de negócios, podem ficar na fase de pré-desenvolvimento. Processos como definição dos requisitos do produto, projeto informacional, projeto detalhado, preparação para produção e lançamento do produto, podem fazer parte da macrofase desenvolvimento. E por fim, processos de acompanhamento e monitoramento, melhorias no produto, pós-vendas, fazem parte do pós-desenvolvimento. O presente trabalho irá tomar como base o modelo de referência proposto por ROZENFELD *et al.* (2006), na macrofase de desenvolvimento, onde, conforme exposto a seguir, contém importantes direcionadores para a velocidade na entrega dos produtos e tratá-los de forma contínua, poderá trazer benefícios, tornando-se um diferencial competitivo para a empresa.

Para CANKURTARAN *et al.* (2013), existem alguns direcionadores para velocidade na entrega de produtos, entre eles, objetivos claros do projeto, liderança efetiva, experiência e dedicação da equipe e a integração. O referido autor trata ainda que a velocidade por si só não é um critério de sucesso, mas quão impactante ela é para o sucesso no desenvolvimento de novos produtos. Custos de desenvolvimento, tempo de introdução no mercado, qualidade do produto e superioridade do produto, tem implicações cruciais para obter resultados favoráveis baseados no cliente e resultados financeiros. Finalmente, em vez de aumentar a produtividade reduzindo erros dispendiosos, os esforços de aceleração podem de fato aumentar sua ocorrência através da eliminação de etapas críticas do processo. A visão predominante sobre a velocidade, o link do sucesso financeiro é um relacionamento sinérgico: velocidade mais rápida aumenta o sucesso financeiro. Como o desenvolvimento rápido de software aumenta a probabilidade de uma empresa entrar em um mercado com menos concorrentes, ela pode seguir uma estratégia de precificação que afeta positivamente os lucros.

### 2.3 - PROCESSO DE DESENVOLVIMENTO DE SOFTWARE E AGILIDADE

Para PRESSMAN (2011), processo de software é definido como uma metodologia para as atividades, ações e tarefas necessárias para desenvolver sistemas com qualidade, definindo a abordagem adotada conforme o mesmo é elaborado pela engenharia. O processo de engenharia de software é formado por um conjunto de passos de processo parcialmente ordenados, relacionados com artefatos, pessoas, recursos, estruturas organizacionais e restrições, tendo como objetivo produzir e manter os produtos de software finais requeridos. As tarefas do processo de desenvolvimento de software são organizadas dentro de uma metodologia a ser adotada por cada empresa. Ainda para o mesmo autor, referente ao processo, são estabelecidas cinco atividades metodológicas, a saber: comunicação, que tem a intenção de compreender os objetivos das partes interessadas, que ajudarão a definir as funções e características do software; planejamento, onde cria-se um mapa que ajuda a guiar a equipe no desenvolvimento da solução, contendo informações como atividades, riscos, recursos e produtos resultantes a serem produzidos; modelagem, onde cria-se um “esboço” do software, de modo que se possa ter uma ideia do todo; construção, que combina a geração de código e testes necessários para revelar erros na codificação; e entrega, onde o software, como uma



entidade completa ou como um incremento parcialmente efetivado, é entregue ao cliente, que avalia o produto entregue e fornece feedback, baseado na avaliação.

A maneira pela qual os projetos de desenvolvimento de software executam as atividades e as fases de sua metodologia, seja de forma sequencial, interativa, incremental ou paralela, são a base para classificação dos projetos entre preditivos ou adaptativos. A escolha entre um ou outro está relacionada a natureza do produto a ser desenvolvido. Quando o escopo do produto é bem conhecido, definido no começo do projeto e não requer muitas mudanças durante sua execução, opta-se pelo uso de projetos preditivos.

Nos modelos preditivos, o responsável pelo projeto coleta todas as informações do que deve ser produzido no início do mesmo, planeja, define todas as atividades e recursos necessários para produzi-las antes de passar para fase de execução. Essa necessidade de documentação e levantamento dos dados no começo do projeto, visando seguir conforme o planejado, tornou o modelo conhecido como modelo baseado em plano (*plan-based*) e tem ainda como característica entregas em grandes lotes.

Os modelos adaptativos estão relacionados a produtos inseridos no cenário de incerteza e grande volume de mudanças ao longo de sua execução. Devido a isso, pretendendo a redução de desperdícios ou entrega de requisitos não mais necessários pelo cliente, os responsáveis pelo projeto realizam um planejamento macro do produto e um planejamento mais apurado somente do que será entregue nas próximas semanas. Os modelos adaptativos são também chamados de modelos baseados em valor, estando assim alinhados com as metodologias ágeis de desenvolvimento de software.

Conforme explanado em DYBA e DINGSOYR (2008), os processos de desenvolvimento ágil de software abordam o desafio de um mundo imprevisível, contando primeiramente com pessoas e sua criatividade, em vez de processos. Para os autores, agilidade é vista como a remoção do máximo de peso, comumente associado às metodologias de desenvolvimento baseado em plano, quanto possível para promover uma resposta rápida às mudanças de ambientes, mudanças de requisitos do usuário, prazos de projetos acelerados, entre outros. As metodologias ágeis são desenvolvidas assim para abraçar, em vez de rejeitar, taxas de mudanças mais altas nos produtos. Em pesquisa realizada por MELO *et al.* (2013), demonstrou que o método ágil mais seguido pelas empresas no Brasil é o Scrum, sucedido pela combinação de Scrum com XP (*eXtreme Programming*), modelos híbridos e Scrumban, *Lean development* e outros métodos como *Test-Driven Development* (TDD), *Feature-Driven Development* (FDD) e uso do Scrum combinado com o PMBOK. Além disso, o autor trouxe como as principais práticas

adotadas nos métodos ágeis o planejamento de iteração, realização de retrospectivas, teste de unidade, reuniões diárias e refatoração de código. Essas práticas representam gestão, melhoria contínua, qualidade e aspectos arquitetônicos valorizados nas equipes.

Nesse sentido, empresas estão adotando metodologias ágeis de desenvolvimento de produtos, como o Scrum, para a gestão e controle em ambientes que mudam rapidamente. Conforme tratado em CERVONE (2011), a gestão de projetos com Scrum é intencionalmente iterativo e incremental, com foco na equipe. Essa abordagem ajuda a controlar o caos que pode resultar de interesses conflitantes dentro das partes interessadas do projeto, além de permitir uma melhor comunicação, maximizar a cooperação e proteger a equipe de interrupções e impedimentos. Para o autor, o objetivo é entregar um produto mais adequado e mais rapidamente do que os métodos tradicionais. Dentro do Scrum, existem três papéis principais para sua realização: *Product Owner* (Dono do Produto), *Scrum Master* e Time Scrum. O Dono do Produto representa a voz do cliente, principal responsável por identificar as necessidades do mesmo e transformar em requisitos técnicos o qual o produto a ser desenvolvido precisa conter. Sendo assim, o dono do produto é o papel fundamental para alinhar o trabalho da empresa, com o que o produto precisa entregar para apoiar a estratégia. O Scrum Master tem como característica ser um líder servidor, removendo impedimentos para que o time scrum trabalhe de forma efetiva. O Time Scrum produz os entregáveis do projeto, sendo estes validados pelo *Product Owner*.

Apesar da adoção dessas metodologias ágeis, o processo de desenvolvimento de software continua presente, mesmo que de forma informal dentro do dia a dia da equipe. A equipe precisa ainda planejar, coletar requisitos, analisar, documentar, codificar, testar, homologar, validar, publicar para a entrega final do produto. Ou seja, a gestão ágil dos processos de desenvolvimento é apenas parte do processo ponta a ponta da entrega de produtos, sendo necessário assim o uso de outras abordagens, que envolvam outros setores da empresa em prol da agilidade dos negócios e atingimento dos objetivos estratégicos.

## 2.4 - INTEGRAÇÃO CONTÍNUA

Para BRAGA (2015), integração contínua é uma prática de desenvolvimento de software, que consiste em um processo de integrar continuamente o trabalho desenvolvido, em um repositório com o resto da equipe de desenvolvimento executando

os testes do trabalho integrado a cada *checkin* (envio de código fonte do sistema) no repositório, permitindo assim detectar e localizar erros rapidamente. Para STAHL e BOSCH (2017), os termos integração contínua (*Continuous Integration*) e entrega contínua (*Continuous Delivery*) são utilizados para designar qualquer sistema de atividades automatizadas, realizadas para transformar o código-fonte em produtos funcionais e potencialmente entregáveis e implantáveis com qualidade, conteúdo e funcionalidade conhecidos, incluindo compilação, vinculação, empacotamento, teste, geração de documentação e muito mais, servindo para garantir que o software possa ser liberado para produção a qualquer momento. Segundo SHAHIN *et al.* (2017), integração contínua é uma prática de desenvolvimento amplamente estabelecida na indústria de desenvolvimento de software, no qual membros de uma equipe integram e mesclam o trabalho de desenvolvimento com frequência, por exemplo, várias vezes ao dia. Tal característica, visa permitir que as empresas tenham um ciclo de lançamento mais curto e frequente, melhore a qualidade do software e aumente a produtividade de sua equipe. Para o autor, a integração contínua é uma base para a Entrega Contínua, no qual a implementação de práticas e ambientes confiáveis e estáveis de Integração Contínua deve ser a primeira e mais alta prioridade para uma determinada organização adotar com sucesso a prática da Entrega Contínua.

Organizações que não usam integração contínua possuem longos períodos entre as integrações fazendo com que seu processo de rastreabilidade de localizar e resolver problemas aumentem consideravelmente. A Figura 2.2 apresenta as partes de um sistema de integração contínua, onde os desenvolvedores primeiramente colocam os seus trabalhos de código-fonte no repositório de controle de versão de software, de onde o integrador de solução irá verificar as mudanças no repositório e irá executar o *script* de integração com o software principal, gerando feedbacks caso algo de errado aconteça.

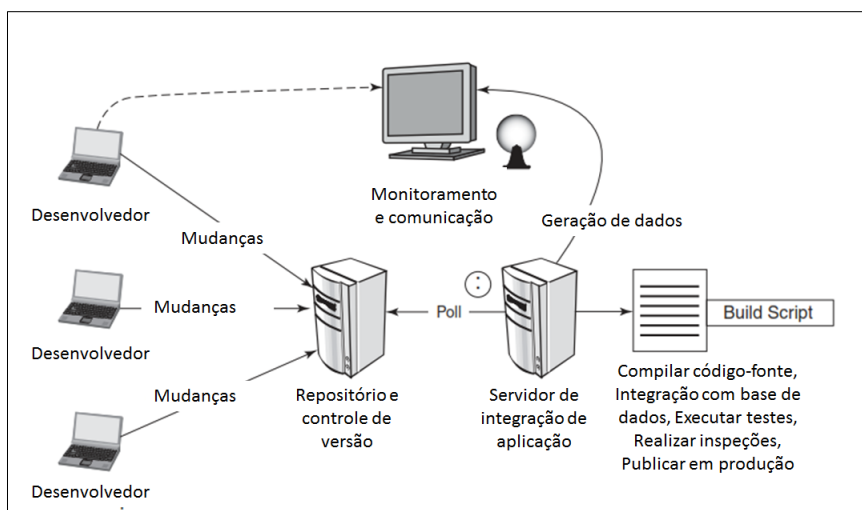


Figura 2.2 - Componentes de um sistema de integração contínua.  
 Fonte: DUVALL (2007).

Segundo BRAGA (2015), para usar integração contínua, existem alguns princípios e práticas chaves que são usadas por algumas organizações, dentre elas o uso de controle de versão com repositório único, emprego de *build* automatizado, executar testes próprios, executar *commits* diários no repositório principal, usar servidor de integração e executar um conjunto de testes automatizados e abrangentes. Nesse sentido o seu uso permeia comumente por alguns setores técnicos da empresa, a saber: analista de sistemas e desenvolvedores de software, equipe de arquitetura, equipe de teste de software e equipe de produção. Para DUVALL (2007), muitas equipes acham que essa abordagem leva a problemas de integração significativamente reduzidos e permite que uma equipe desenvolva software coeso mais rapidamente. Integração contínua é realmente promover responsabilidade e fornecer valor ao cliente, tendo uma imagem muito maior em mente: uma equipe responsável que garante que os novos recursos possam ser enviados à produção e ao cliente rapidamente, e que garante que todos os membros possam realizar seu trabalho e verificar os novos códigos, garantindo que essa mudança não passe apenas na compilação mas que também tenham um impacto mínimo no ambiente de produção (MEYER, 2014).

Apesar de sua importância para redução de problemas na integração de software, o tema integração contínua começou a ser ligado mais fortemente ao ambiente de negócios da empresa com o advento do movimento DevOps (*Development and Operations*), impulsionado pelo uso de práticas ágeis de desenvolvimento e gestão de infraestrutura como código. Usando a abordagem ágil, o time de desenvolvimento tende a produzir novas versões do produto a cada duas ou quatro semanas. Estando os usuários

finais envolvidos nesse desenvolvimento, poderão garantir um feedback rápido, inspirando assim mudanças mais rápidas. Devido a isso, empresas tem adotado em sua infraestrutura tecnologias como: ferramentas de gerência de configuração, integração contínua de código, técnicas de publicação de código em produção, virtualização de servidores e uso de computação na nuvem, que visam permitir uma gestão mais eficiente do processo de desenvolvimento de produto, envolvendo pessoas, hardware e software, que pretendem promover uma escalabilidade do ambiente de forma segura e transparente para os desenvolvedores e clientes.

Para KIM *et al.* (2016), com a introdução do DevOps e a comoditização de hardware e software, os recursos podem ser criados em semanas, sendo rapidamente implantados na produção em apenas horas ou minutos - para essas organizações, a implantação finalmente se tornou rotina e baixo risco. Essas organizações são capazes de realizar experimentos para testar ideias de negócios, descobrindo quais ideias criam mais valor para os clientes e para a organização como um todo, que são posteriormente desenvolvidas em recursos que podem ser implementados de maneira rápida e segura na produção. Onde, em uma época em que a vantagem competitiva exige tempo de colocação no mercado e experimentação incansável, as organizações que são incapazes de replicar esses resultados estão destinadas a perder no mercado para concorrentes mais ágeis e podem potencialmente sair do negócio completamente. O relatório “*State of DevOps 2018*”, que nos últimos sete anos, pesquisou mais de 30.000 profissionais técnicos em todo o mundo para explorar as relações entre desempenho de TI, práticas de DevOps, cultura, desempenho organizacional e outros elementos que afetam os resultados dos negócios, demonstram que as empresas altamente evoluídas tem uma cultura DevOps que abrange vários departamentos, automatizou mais serviços para uso amplo (além do âmbito de uma equipe), automatizou mais medições para os objetivos de negócios e compartilha padrões e melhores práticas de maneira ampla.

Considerando a necessidade de entregar produtos evoluídos constantemente, com agilidade e segurança, torna-se imperativo ter um processo de desenvolvimento de produto devidamente alinhado com a estratégia de negócios da empresa, onde a sua execução seja feita utilizando ferramentas de integração contínua que permitam validar de forma rápida se um produto irá entregar o resultado almejado ou não, sem prejudicar o que já está em uso pelo cliente.

## **CAPÍTULO 3**

### **MATERIAIS E MÉTODOS**

#### **3.1 - MATERIAIS**

Para atingir os objetivos propostos e testar as hipóteses formuladas, realizou-se um estudo utilizando dados e informações oriundas do sistema de gestão organizacional, com foco em dois ambientes.

O primeiro ambiente está relacionado ao sistema de gestão da qualidade da empresa, que contém a documentação de todos os processos organizacionais vigentes, tanto os processos primários, quanto os processos de apoio e de gestão. Cada processo possui em sua descrição o objetivo do processo, partes envolvidas no mesmo, indicadores de desempenho, fluxo macro de execução (modelado em sua maioria usando BPMN), descrição de como cada atividade é executada, lista de documento necessários para execução do processo.

O segundo ambiente é o sistema de gestão integrada da empresa, que consiste em um sistema de informação onde os colaboradores da empresa abrem tickets (chamados) entre os setores, para que determinada tarefa seja executada e devidamente registrada para fins de gestão de conhecimento. O referido sistema contém diversas informações, como: cliente, dados do demandante, data de abertura, previsão de atendimento, data de encerramento, assunto, prioridade, entre outros. Através desses chamados que a equipe registra as atividades, a cada interação realizada, até a conclusão do atendimento.

#### **3.2 - CICLO DA PESQUISA**

Esta seção apresenta as principais etapas que constituíram o ciclo desta pesquisa, com uma breve descrição e associação entre as etapas.

- Planejamento da pesquisa: esta etapa iniciou-se com a definição do processo a ser avaliado (formalizado ou não pela empresa), focando no processo de desenvolvimento de produto de software e desenvolvimento de sistema de informação, e todas as suas possíveis fases dentro da organização, utilizando os sistemas de gestão da empresa para captura dos dados, bem como com a definição dos itens de medição: tempo de execução de uma atividade de publicação de

nova versão do produto, tanto em ambiente de homologação quanto em ambiente de produção. Ou seja, desde a necessidade até a sua disponibilidade para o cliente. Identificando ainda o processo que gera a necessidade de publicação de código fonte no ambiente de produção.

- Coleta dos dados: a coleta foi do tipo quantitativa, onde nessa fase foram coletados os dados disponíveis no sistema de gestão empresarial, dos anos de 2019 e 2020, o qual são registrados através de “chamados” (tickets), que representam um tipo de pedido de execução de um processo por uma equipe. Esses tickets podem se tornar projetos, dependendo da amplitude das atividades a serem realizadas por ele. O tipo de chamado avaliado foi do tipo “Deploy de sistema em produção”, atividade essa que é a publicação de uma evolução do produto de software no ambiente de produção, envolvendo desde o a disponibilização do código em um local e sua cópia pelo setor responsável ao servidor de produção. Foi capturado o tempo do ciclo da abertura do chamado até sua conclusão, bem como a quantidade de interações no processo, conforme apresentado na Tabela 3.1.

Tabela 3.1 - Modelo de tabela de coleta de dados.

Nº Chamado	Data de Abertura	Data de Fechamento	Tempo de Espera	Quantidade de interações	Tempo de Execução
------------	------------------	--------------------	-----------------	--------------------------	-------------------

- Análise dos dados: após coleta dos dados, foi feita uma análise quantitativa dos principais chamados e dificuldades encontradas nas atividades de integração contínua dentro do processo de desenvolvimento de software existente na organização. Ou seja, o modelo “AS-IS” na modelagem de processos de negócios.
- Proposta de Adaptação do Processo de Integração Contínua: foi feita uma proposta, o modelo “TO-BE”, utilizando a notação BPMN, de um novo processo de desenvolvimento de produto de software, promovendo o alinhamento estratégico das entregas com o processo de desenvolvimento de sistemas da organização.

Na Figura 3.1 é possível visualizar o desenho da pesquisa, identificando suas fases.

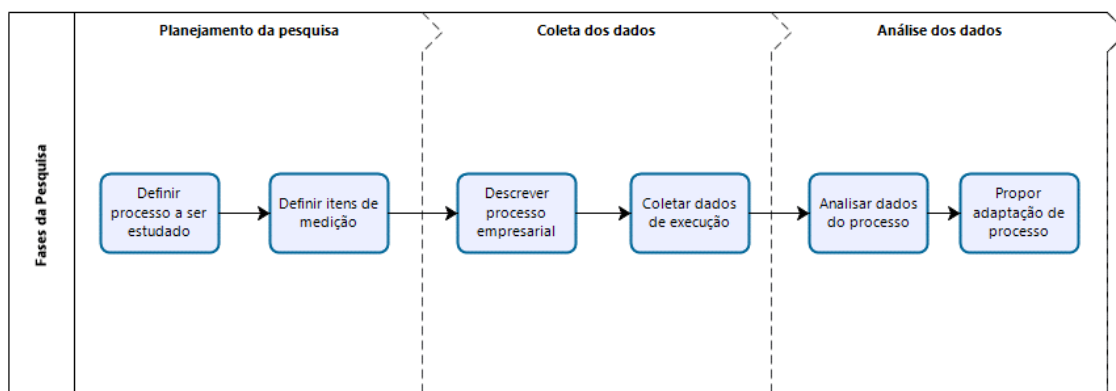


Figura 3.1 - Descrição das fases de pesquisa.

### 3.3 - FERRAMENTAS UTILIZADAS

A partir dos dados coletados foi utilizada a ferramenta da qualidade Gráfico de Controle para evidenciar a variação das ocorrências de tempo de publicação de código-fonte em homologação e produção, para criar uma base de comparação. Para desenho do processo de negócio, foi usado o BPMN – Notação para Modelagem de Processo de Negócio - uma notação para representação visual das principais atividades do processo de desenvolvimento de produtos da empresa.

### 3.4 - SOFTWARES UTILIZADOS

Foram utilizados os softwares Microsoft Office Excel, para tabulação dos dados, realização de análise dos valores coletados, utilizando a função de Tabela Dinâmica para sumarização dos resultados. Utilizou-se ainda a ferramenta Bizagi Modeler (<https://www.bizagi.com/pt>) para desenho dos processos de negócio.



## **CAPÍTULO 4**

### **RESULTADOS E DISCUSSÃO**

#### **4.1 - CARACTERIZAÇÃO DA EMPRESA**

A Companhia de Processamento de Dados é uma empresa de economia mista de capital fechado vinculada à Secretaria de Administração e Gestão, cujo maior acionista é o Governo do Estado.

Hoje, a Empresa tem como objetivo a prestação de serviços especializados em Tecnologia da Informação e Comunicação aos órgãos integrantes da Administração Pública Estadual, podendo, complementarmente, atender órgãos federais, instituições privadas, administrações municipais e outras administrações estaduais. Foi ainda a primeira, na administração pública estadual, a obter a certificação norma ISO 9001:2015. A empresa tem como principais serviços a oferta de sistemas para a Secretaria de Educação, Departamento Estadual de Trânsito e Secretaria de Administração e Gestão. Possui ainda em seu portfólio outras linhas de produto, tais como: infraestrutura para hospedagem de sistemas e aplicativos, conectividade entre órgãos e desenvolvimento de software sob demanda. A empresa conta com um quadro de aproximadamente 390 funcionários das quais aproximadamente 300 são dedicadas às atividades de TI, sendo que aproximadamente 150 colaboradores atuam em projetos de desenvolvimento de software. Vale salientar que o desenvolvimento de software acontece tanto na sede da empresa quanto em órgãos que possuem um contrato de termo de cessão de pessoal. O referido trabalho irá tratar unicamente do desenvolvimento realizado na sede da empresa.

Ser reconhecida como a melhor opção de provedor de serviços digitais para a gestão pública e à iniciativa privada no estado do Amazonas, e ser autossustentável, é a missão da empresa. Para isso, a mesma precisa contar com processos de gestão devidamente alinhados e ágeis.

##### **4.1.1 - Descrição do processo de gerenciamento de portfólio de produtos**

A empresa possui o processo de gerenciamento de portfólio de produtos que tem como objetivo gerenciar os serviços disponibilizados no catálogo de serviços da organização, garantindo que este seja uma fonte íntegra de informações, fornecendo

detalhes de cada serviço e componentes. O referido processo tem como principal processo fornecedor a identificação de necessidade do cliente, que trata das ações de identificação de oportunidade de negócios, e como saída a definição de produto/serviço devidamente documentada.

Conforme visualizado na Figura 4.1, o processo de criar produto se inicia com a necessidade de modelagem de um novo produto identificada, permeia pela elaboração do documento de descrição do produto, elaboração da planilha de custo e precificação, onde a partir desses dados é feita a análise de viabilidade para que se decida prosseguir com a criação desse produto ou não. O referido processo finaliza quando esse produto é criado no portfólio de produtos da organização. Nesse cenário, não é feito o acompanhamento para garantir se os requisitos relacionadas a esse produto de fato estão sendo produzidos.

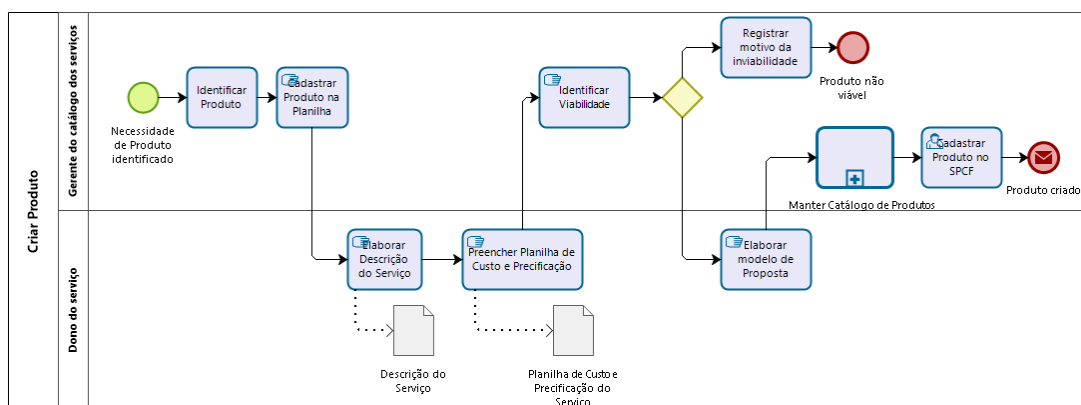


Figura 4.1 - Processo criar produto.  
Fonte: SGQ da organização.

Na atividade “Elaborar Descrição do Serviço” o dono do serviço preenche o documento de descrição do serviço com informações relacionadas a ele, tais como: nome do serviço, característica do serviço, composição, serviços opcionais, local de prestação de serviço, elementos que compõe o serviço, o que não está no escopo, responsabilidades do cliente, entre outras informações. Em “característica do serviço”, estão os principais dados sobre o que o serviço irá conter e ser entregue para os clientes. A atividade “Preencher planilha de custo e precificação”, é responsável por armazenar informações que serão utilizadas como base para elaboração do preço do serviço, tais como: aquisições, contratos mensais, recursos de infraestrutura de armazenamento, recursos humanos, oportunidades de mercados, produtos similares, volumetria e precificação do produto. O referido processo apoia então a comercialização de produtos na empresa,

gerando informações de como o produto funciona e qual o preço a ser praticado em sua comercialização.

Em 2020, ano da implantação do referido processo, observou-se que aproximadamente 88% dos produtos criados foram relacionados a produtos que envolvem desenvolvimento de sistemas, tornando assim o processo de desenvolvimento de sistema na organização bastante estratégico.

#### 4.1.2 - Descrição do processo de desenvolvimento de sistemas

A empresa possui um processo de desenvolvimento de sistemas que tem como objetivo fornecer orientações para a realização de atividades técnicas que ocorram no contexto de projetos de desenvolvimento e de manutenção de sistemas de informação desenvolvidos pela organização. O referido processo tem como processo fornecedor “Gerenciar Projeto”, que trata das ações de viabilidade e execução de projetos e como saída o produto de software propriamente dito.

Considerando a integração contínua como um conjunto de práticas aplicado diariamente no time de desenvolvimento de software, a coleta de dados teve como foco a atividade de “Deploy de sistema em produção”, que faz parte do processo de desenvolvimento de sistemas implantado. Segundo o sistema de gestão da qualidade da organização, o desenvolvimento de sistemas é realizado em ciclos conforme planejamento operacional, na qual cada ciclo representa uma entrega potencial ao cliente. Na Figura 4.2, visualiza-se o processo de desenvolvimento de sistemas vigente da empresa.

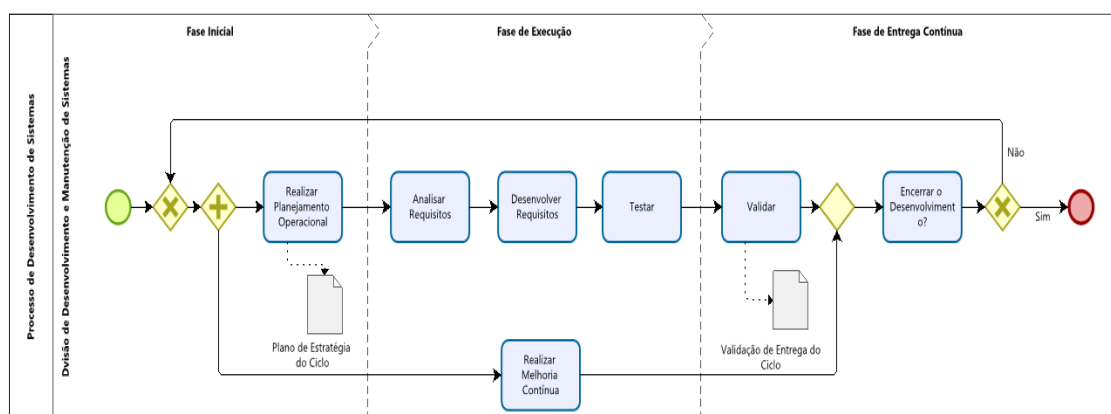


Figura 4.2 - Processo desenvolvimento e manutenção de sistemas.

Fonte: SGQ da organização.

Na fase inicial do processo, tem-se a atividade “Realizar Planejamento Operacional”, onde se reavalia a capacidade de atendimento da demanda pelo time e o tamanho do ciclo de entregas é definido. Durante a fase de execução, tem-se a atividade “Analisar requisitos”, onde são detalhados, documentados e aprovados os requisitos do software solicitado. Na atividade “desenvolver”, é construído em ambiente de desenvolvimento, testado e avaliado cada componente de software solicitado. Na atividade de “Testes”, os testes são realizados visando identificar a qualidade do software conforme solicitado. A “fase entrega contínua”, uma das características da integração contínua, está representada como o último passo do processo. Antes disso, a atividade “Validar”, tem como objetivo a validação e documentação dos resultados junto ao cliente, disponibilizando o serviço/produto ao cliente conforme acordos de entrega. Essa validação pode ser feita tanto em ambiente de homologação como em ambiente de produção. A atividade “Encerrar desenvolvimento” visa identificar a necessidade de prosseguir com mais alguma interação ou concluir a demanda.

A empresa faz uso de um repositório único para armazenamento de código fonte e controle de versão dos sistemas desenvolvidos, chamado de Git. Git é um sistema de controle de versão distribuído gratuito e de código aberto projetado para lidar com tudo, desde projetos pequenos a muito grandes com velocidade e eficiência. O objetivo do repositório de controle de versão é gerenciar alterações no código-fonte e outros ativos de software (como documentação) usando um repositório de acesso controlado, fornecendo assim um "único ponto de origem" para que todo o código-fonte esteja disponível em um local principal. O repositório de controle de versão permite ainda que você “volte no tempo” e obtenha diferentes versões do código-fonte e outros arquivos.

Para que o trabalho de desenvolvimento de sistemas seja realizado de forma sincronizada utilizando o repositório único, a empresa elaborou um fluxo de trabalho para o desenvolvimento e entregas de um projeto com um ciclo básico baseado nas fases de desenvolvimento, qualidade, revisão e lançamento. Segundo a organização, o desenho do fluxo de trabalho é orientado para equipes ou empresas que integram um processo de controle de qualidade como parte de seu ciclo de desenvolvimento, bem como na prestação de um servidor de homologação (*stage*) estável para o cliente avaliar e aprovar o lançamento (*release*) pendente. A Figura 4.3 representa o fluxo de trabalho na ferramenta Git, em uso pela empresa nesse estudo de caso. Nela, as fases do ciclo de todo o desenvolvimento e lançamentos são divididos em objetivos, referidos como “*milestones*”. Nota-se que um *milestone* não necessariamente equivale a um lançamento

(*release*). Uma versão de um projeto poderia ser composto de várias etapas, de acordo com o planejamento do projeto e os recursos disponíveis. Essas fases são representadas por *branches* "permanentes" e os "temporários" no repositório git, que facilitam o processo de desenvolvimento por meio da lançamento (*release*). Segundo a empresa, a principal razão para a existência de *branches* persistentes é prover o deploy multifacetado, permitindo qualquer um de ver o estado de uma aplicação em diferentes estágios de desenvolvimento.

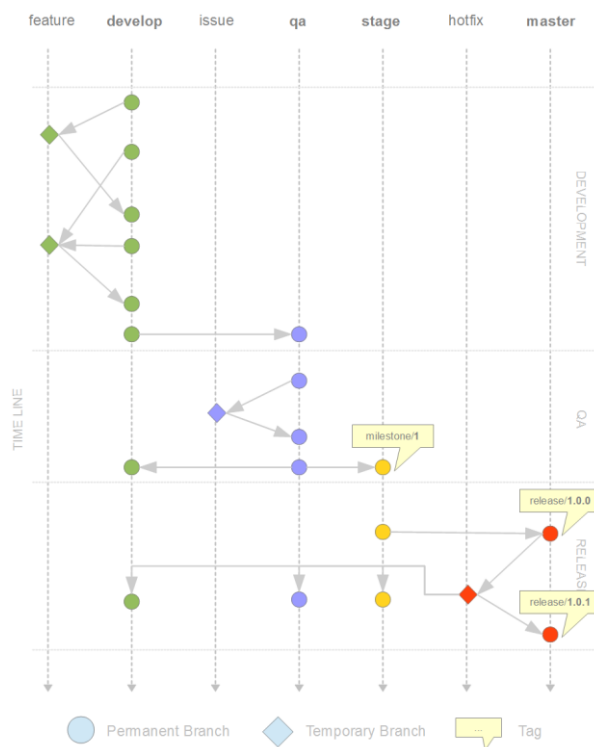


Figura 4.3 - Fluxo de trabalho Git.  
Fonte: SGQ da organização.

Para que todas as evoluções do sistema sejam aplicadas em ambiente de homologação e produção, foram criadas os tipos de chamados: “Deploy de Sistema em Homologação” e “Deploy de Sistema em Produção”. Esses chamados são direcionados para execução da equipe de infraestrutura, onde, qualquer alteração em um desses ambientes requer a abertura de um novo chamado para execução do serviço.

#### 4.2 - CARACTERIZAÇÃO DO PROBLEMA

Apesar da existência de um processo de desenvolvimento de sistemas na empresa, o mesmo representa apenas atividades relacionadas ao time de desenvolvimento (analista

de sistemas, programadores, designer, teste), onde, a interação com outros times é feita através da realização de chamados. O problema identificado e analisado, primeiramente, foi o grande volume de “chamados” abertos para a equipe de infraestrutura, solicitando a publicação de código fonte gerado no processo de desenvolvimento de sistemas, tanto em ambiente de homologação como em ambiente de produção. Ao analisar dados disponibilizados pelo sistema de gestão da organização, observou-se que em 2019 houve um total de 905 pedidos de publicação e em 2020 essa quantidade reduziu para 622 chamados, conforme resultado apresentado na Tabela 4.1. A motivação principal para redução desse número foi que o processo de desenvolvimento de alguns sistemas começaram a adotar práticas de integração contínua.

Tabela 4.1 - Total de chamados por tipo/ano.

<b>Total de Chamados/Ano</b>	<b>2019</b>	<b>2020</b>	<b>Total por tipo</b>
Deploy de Sistema em Homologação	221	207	428
Deploy de Sistema em Produção	684	415	1099
<b>TOTAL GERAL</b>	<b>905</b>	<b>622</b>	<b>1527</b>

Para a realização das atividades previstas nos chamados, é necessário a interação de diversos técnicos, onde cada passo é registrado no sistema de gestão integrada, desde a criação do chamado até ações como: assumir o chamado, encaminhar para grupo solucionador, novo apontamento registrado, concluir tarefa, concluir chamado, e aprovação do chamado pelo demandante. A quantidade de passos dentro de cada chamado varia entre 3 até 30 passos para conclusão do serviço, podendo ainda cada passo desse ser realizado por uma pessoa diferente dentro da empresa. A variação do número de pessoas que interagem no chamado é de 2 até 5, para os chamados aqui listados, envolvendo o setor demandante (Desenvolvimento de sistemas) e o setor demandado (Infraestrutura de servidores).

Conforme conversado com o Gerente de Infraestrutura da organização, apesar do tempo de execução do processo de deploy ser baixo, podendo durar de 10 minutos até 4h, o tempo de espera para execução é alto, variando de 4h a até 9 dias de espera para início da execução do serviço, dependendo do volume de demandas recebido pelos técnicos responsáveis.

Além do informado, temos ainda como problemática o não alinhamento dos chamados abertos pela equipe de desenvolvimento com a estratégia organizacional. Ou seja, os chamados executados pela equipe de infraestrutura não possuem explicitamente uma motivação para realização das atividades, ou um produto associado a ele. Muitos podem até contribuir para os ganhos globais da empresa, mas isso não está sendo representado dentro do processo de publicação do produto, onde, o pedido feito é apenas para que o técnico publique o código fonte em produção, copiando o mesmo de uma base previamente informada no chamado de origem. O referido processo não envolve ninguém da área de negócios da organização, seja um analista de negócios ou a área de gestão de portfólio de produtos. Todo esse cenário provoca constantemente quebra no relacionamento com o cliente, tanto pela demora na entrega dos produtos, entrega de incremento fora do escopo, ou falha na comunicação do andamento da demanda.

A ideia de integração dos processos de criação de produto e desenvolvimento de sistemas, já alinhado com a integração contínua, foi proposta inicialmente pelo escritório de projetos da organização, o qual necessitava acompanhar se as entregas feitas pelas equipes técnicas estavam dentro do escopo contratual, além do custo e prazo acordado. Infelizmente a consolidação do escritório de projetos, e conseqüentemente a gestão de portfólio de projetos, não aconteceu, sendo necessário o uso de uma outra abordagem baseada em produtos, e não mais em projetos.

E com isso teve a iniciativa de propor a melhoria nas entregas em dois produtos de software, com a execução de processos, mesmo de que forma informal, que envolvessem a formação de times multifuncionais, com pessoas das mais diversas áreas da organização, primeiramente: Divisão de Produtos, Divisão de Desenvolvimento de Sistemas e Divisão de Infraestrutura. A dinâmica assim foi modificada, onde o Dono do Produto, presente na divisão de produtos, reportava o avanço das entregas para o setor de negócios e este contribuía para a priorização e gestão dos requisitos. Além disso, com pessoas da infraestrutura compondo o time, os mesmos já estavam cientes de como o trabalho deles poderia contribuir para a estratégia da organização.

O maior problema percebido durante esta estratégia foram dois pontos. O primeiro ponto foi a definição das responsabilidades e suas fronteiras, visto que, a empresa vinha de uma transição do papel de gerente de projetos que não foi devidamente efetivada, sem saber quem seria o responsável pelo gerenciamento da lista de entregas para o cliente e quem seria responsável por produzir essas entregas. A segunda foi saber se essa forma de trabalho poderia ser escalada para todos os produtos de software geridos pela organização,

sendo necessário a mudança de cultura perante os funcionários e demais gestores da empresa, e um apoio significativo da alta gestão.

#### 4.3 - PROPOSTA DE PROCESSO DE DESENVOLVIMENTO DE PRODUTO DE SOFTWARE

##### 4.3.1 - Processo atual

O processo atual de desenvolvimento de produto de software, conforme apresentado nos tópicos 4.1.1 e 4.1.2, é feito de forma dividida e sem conexão entre os dois processos citados. O primeiro trata somente de elaborar modelo de negócio e precificação do produto a ser produzido, não garantido sua entrega. O segundo processo trata do desenvolvimento de sistemas, sem a integração com o setor de infraestrutura e com o setor de negócios, dificultando assim a realização de entregas estratégicas, na velocidade almejada pela empresa.

A empresa tratada nesse estudo de caso não apresenta assim um processo formal, padronizado, de desenvolvimento de produtos de software de forma integrada, conforme modelo de referência proposto por ROZENFELD.

##### 4.3.2 - Processo proposto de desenvolvimento de produto de software

O número de requisitos e mudanças evolutivas/corretivas em produtos está crescendo na empresa, hoje ela conta com aproximadamente 30 sistemas e vários módulos em operação, utilizados por seus clientes, nas áreas de educação, trânsito, recursos humanos, segurança pública, entre outros. As linhas de produtos relacionada a desenvolvimento de sistemas, execução de sistemas e software como serviço, são responsáveis por 60% do faturamento mensal da empresa.

A proposta é definir um processo conforme mostra a Figura 4.4, sendo realizado pela integração de quatro grandes áreas da organização, com início e fim junto a área de negócios (dono do produto). Os parágrafos seguintes, temos as informações do processo proposto, tais como, objetivos, indicadores e descrição das atividades. A Figura 4.4 ilustra o processo de desenvolvimento de produto de software. O Anexo I contém a mesma Figura, mas de forma ampliada, para melhor visualização.



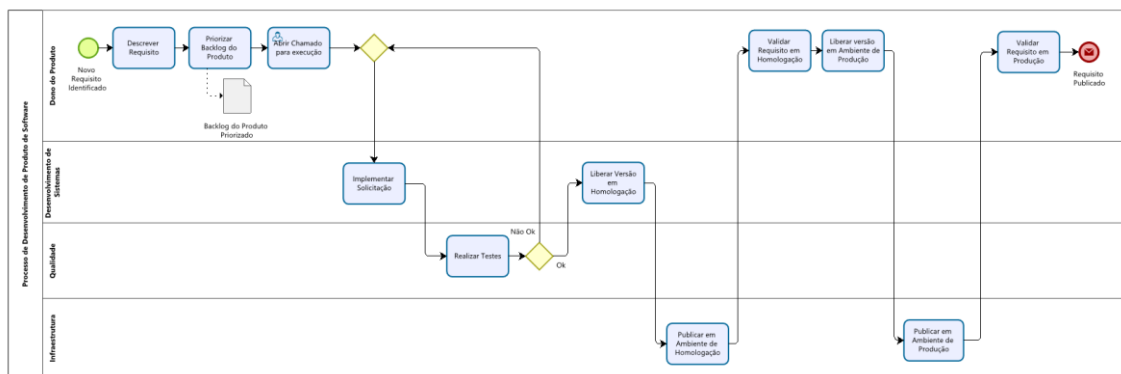


Figura 4.4 - Processo desenvolvimento de produto de software.

Objetivo: Responsável por criar um incremento de software, garantindo que o mesmo esteja alinhado com os objetivos da organização, fornecendo detalhes de cada evolução e seus componentes.

#### 4.3.2.1 - Matriz do processo por área

- Apoio: Toda a organização;
- Cliente: Toda a organização;
- Fornecedor: Toda a organização;
- Responsável: Divisão de Produtos, Gerência de Negócios.

#### 4.3.2.2 - Mapa geral do processo

A Tabela 4.2 apresenta o mapa geral do processo, informando os processos fornecedores, fontes de entrada e as saídas do processo Desenvolvimento de Produto de Software.

Tabela 4.2 - Mapa geral do processo.

Processo Fornecedor	Entrada	Saída
Gerenciamento de Mudança	Planejamento Estratégico	Definição de Serviço Documentada
Gerenciamento de Incidentes	Requisitos novos de negócios	Atualização do Catálogo de Serviços

Gerenciamento de Configuração	Necessidades atuais e futuras do portfólio de Serviços	Revisar Catálogo de Produtos
Identificação da Necessidade de Cliente	Necessidade Identificada	Revisar Catálogo de Produtos

#### 4.3.2.3 - Indicadores

A Tabela 4.3 apresenta os indicadores do processo, informando o indicador, unidade de medida, responsável, fórmula, objetivo e meta do processo Desenvolvimento de Produto de Software.

Tabela 4.3 - Indicadores do processo.

<b>Indicador</b>	<b>Unidade de medida</b>	<b>Responsável</b>	<b>Fórmula</b>	<b>Objetivo</b>	<b>Meta</b>
Frequência de Entrega	Quantidade	Divisão de Produtos	Quantidade de Requisitos Entregues em um período de tempo	Tornar previsível a entrega de requisitos	A definir
Capacidade de Entrega	Unidade	Divisão de Produtos	Quantidade de requisitos possíveis de serem entregues em um período de tempo	Tornar visível a capacidade de entrega do time, permitindo um melhor planejamento	A definir

Atingimento do Resultado	Percentual	Divisão de Produtos	Percentual de contribuição do requisito com o resultado da empresa	Promover o alinhamento estratégico do requisito a ser entregue	A definir
--------------------------------	------------	------------------------	--	---	-----------

#### 4.3.2.4 - Descrição das atividades

– Descrever Requisito: eventualmente, ao identificar-se uma nova necessidade do cliente, seja através de visitas, conversa com analistas de negócios ou com o time de desenvolvimento, o dono do produto recebe a solicitação de novo incremento de produto. A descrição recomendada é utilizando “estórias de usuário”, que contém informações sobre para quem é o requisito, o que será entregue e para que será entregue.

– Priorizar Backlog do produto: O dono do produto, após descrever o requisito solicitado, e conhecendo todos os requisitos existentes em seu backlog (lista de requisitos do produto), decide o que fazer com o novo requisito em sua fila. A priorização deve ser feita sempre considerando o valor a ser entregue ao cliente do produto, usando as técnicas de priorização que for mais conveniente, a saber: Valor X Esforço, MoSCoW, Kano, entre outras. O documento que representa o backlog do produto possui informações como: produto, tema do requisito, item, usuário do requisito, prioridade, situação (a fazer, fazendo e feito).

– Abrir chamado para execução: O dono do produto abre o Ticket para iniciar a implementação do incremento do produto, o mais breve possível. Nesse momento, o dono do produto deve ter conhecimento do que o time de desenvolvimento está implementando no momento e sua capacidade de entrega. O registro é feito na ferramenta de gestão integrada da organização, podendo classificar cada pedido como: evolução ou correção.

– Implementar solicitação: O dono do produto, após a abertura do Ticket, reúne com o time do processo, e explica o que ele espera ser entregue com o incremento, definindo os critérios de aceitação para o requisito. Nesse momento toda a equipe do processo é envolvida, como os responsáveis pela codificação da solução, testes e infraestrutura, para identificar as atividades necessárias para criação do produto. Essa

atividade é concluída quando o time de desenvolvimento considera que o incremento está pronto para ser testado e devidamente publicado nos outros ambientes.

- Realizar testes: Durante a atividade implementar solicitação, a abordagem de teste a ser realizada é definida, para que na fase de realização de testes a mesma seja executada. Conforme a necessidade, será definido um procedimento designando etapas específicas de testes e os tipos de testes que serão feitos. O objetivo é garantir que o software implementou corretamente o requisito solicitado, rastreado segundo os requisitos do cliente, que nesse caso é apresentado pelo dono do produto. Temos como resultado a validação e verificação do requisito ou seu não atendimento.

- Liberar versão em homologação: caso o teste de software esteja “ok”, o time de desenvolvimento cria o versionamento no repositório de controle de versão, para disponibilizar a atualização do software no ambiente de homologação. Seguindo os passos da integração contínua, nesse momento é feito a compilação do código-fonte, integração com a base de dados do sistema e a realização de inspeções para publicação em homologação.

- Publicar em ambiente de homologação: com a definição da versão da atualização do sistema, os responsáveis pela publicação nos servidores de homologação irão criar e configurar o ambiente de infraestrutura para receber a nova versão do código-fonte. Caso seja possível, o time de desenvolvimento pode vir a publicar nesse ambiente sem a interferência da equipe de infraestrutura, trabalhando assim em uma estrutura flexível e segura, em um cenário onde o retorno a versão anterior do sistema deve ser viável.

- Validar requisito em homologação: nessa atividade, o time de desenvolvimento apresenta ao Dono do Produto o incremento no ambiente de homologação, para validação. Caso seja validado, o dono do produto planeja a liberação da versão no ambiente de produção. Caso não seja, é registrado o motivo do não aceite e o requisito volta para o início do processo.

- Liberar versão em ambiente de produção: nessa atividade, o dono do produto informa que a versão está disponível para ser publicada no ambiente de produção, solicitando assim a sua publicação.

- Publicar em ambiente de produção: após a homologação da versão atualizada do sistema, os responsáveis pela publicação nos servidores de produção irão criar e

configurar o ambiente de infraestrutura para receber a nova versão do código-fonte, informando ao dono do produto que a atividade foi realizada.

– Validar requisito em produção: nessa atividade, o Dono do Produto valida o incremento no ambiente de produção.

### **4.3.3 - Modificações propostas através do novo processo**

As duas principais mudanças propostas pelo processo são: alinhamento estratégico e integração contínua. O dono do produto, responsável pela gestão do backlog, é o ator do processo que garante que o time entregue valor continuamente. Nesse novo processo, fica claro que, apesar dele ser o principal responsável, o valor só é entregue no final do processo, permeando assim por todo o time. Além disso, o dono do produto visa atingir os indicadores do processo, que não são meramente de controle, mas sim relacionados a frequência das entregas, capacidade do time e atingimento de resultados. Ou seja, quanto melhor esses indicadores, melhor será o resultado empresarial.

Referente a integração contínua, o processo visa garantir que todos estão inseridos em um mesmo objetivo: entregar valor ao cliente através da implementação de requisitos. Nesse sentido, o planejamento das atividades necessárias para o requisito é feito tão logo o requisito é aprovado, e todos saem da reunião de planejamento sabendo o que cada membro do time deve fazer para que o resultado seja alcançado. Com esse alinhamento, não se faz necessário a abertura de chamados entre os times, bastando que as atividades sejam devidamente registradas no sistema de gestão da empresa, ou no próprio GIT, que cuida da gerência de configuração do sistema a ser produzido. Assim, aquele processo onde a equipe de desenvolvimento simplesmente abre um chamado para a equipe de infraestrutura e esse executa, sem saber o resultado empresarial daquela ação, deixa de existir, gerando economia de tempo no processo, promovendo agilidade.

Além das duas principais mudanças apresentadas, o novo processo de desenvolvimento de produtos promove uma integração entre os times, sendo executados de forma transversal, permeando toda a empresa, cultivando a cultura de gestão por processos, quebrando os silos empresariais.

## CAPÍTULO 5

### CONCLUSÕES E SUGESTÕES

#### 5.1 - CONCLUSÕES

Alta competitividade, incertezas e constante mudanças, direcionam as empresas a buscarem meios de aperfeiçoar a entrega de produtos, de forma ágil. A gestão de processos de desenvolvimento de produtos mostrou-se um importante aliado na melhoria do resultado organizacional.

Conforme descrição do processo macro de integração contínua de software existente na empresa estudo de caso, utilizando a notação BPMN e descrição do processo, observamos que o referido processo não tinha elo transparente com a estratégia organizacional, nem com outros setores que são fundamentais para que a integração contínua funcionasse de fato. Além disso, descrevemos o processo de gestão de portfólio de produtos, o qual observamos que o setor que promovia a gestão de produtos olhava somente para a precificação, sem o envolvimento na entrega do mesmo.

Além das observações do parágrafo anterior, a análise do modelo de processos presentes na empresa de software utilizada como estudo de caso, permitiu quantificar ainda o alto número de chamados abertos para execução de atividades repetitivas e, o não envolvimento da equipe de negócios na validação e entrega de incrementos de software.

Propomos assim, a adaptação do modelo de entrega de produto de software, com abordagem de integração contínua no processo, realizando a integração contínua com a estratégia. O processo de gestão de produtos proposto promoveu a ligação entre os processos de gestão de portfólio do produto e desenvolvimento de sistemas existentes hoje na organização, propondo assim uma nova forma de execução do trabalho. O mesmo poderá fazer parte do sistema de gestão da qualidade, sendo acompanhado através de seus indicadores.

Ao analisar o modelo proposto, observamos que o mesmo contempla as principais partes da organização: negócios, desenvolvimento e infraestrutura. Onde, em entregas executadas usando o modelo proposto, o cliente ficou mais satisfeito, as entregas foram feitas de formas mais assertivas e com qualidade. Pois o cliente tinha como ponto único ponto de contato o Dono do produto, e este era responsável por garantir que as entregas do time estivessem alinhadas com a necessidade do cliente.

A agilidade nos negócios pode ser medida através do tempo de espera do processo, desde o pedido do cliente até sua entrega efetiva. Saber como a empresa funciona, com todas as suas interconexões, é primordial para que esse tempo de espera seja reduzido. A gestão por processo demonstrou quão importante são as interações entre os setores para a entrega de resultados. A gestão de produtos torna evidente como a empresa gerencia e entrega os pedidos do cliente. A integração contínua cria o cenário ideal para que as entregas sejam feitas com qualidade na velocidade que o cliente necessita, de forma segura. Apesar de saber que o processo de desenvolvimento de produtos desenhado, não garante que a empresa irá ter sucesso, o estudo de caso demonstrou que, as responsabilidades pelas atividades ficou mais clara, o monitoramento das entregas foi facilitado, o cliente recebeu melhores retorno sobre o andamento das demandas, a integração entre os setores foi promovida. Ou seja, os produtos onde o processo de gestão de produtos foi aplicado, o resultado empresarial foi aumentado.

A adaptação da metodologia de integração contínua na estratégia de desenvolvimento de produto de software em empresas de economia mista de tecnologia, trouxe o papel do cliente, representado pelo Dono do Produto, para o dia a dia das entregas. O cliente sabe agora o que está em andamento, o que será lançado, o que está pendente. A priorização das demandas permitiu que a empresa entregasse mais valor em menos tempo. A capacidade de entrega ficou melhor gerenciada, ao se conhecer todo o fluxo de trabalho, permitindo assim a tomada de decisões mais assertivas.

## 5.1 - SUGESTÕES

Devido à grande importância do assunto abordado neste trabalho, são apresentadas, a seguir, algumas sugestões para a continuação:

- Realizar um aprofundamento nas atividades existentes no processo, visando identificar o que pode ser melhorado através de automatizações.
- Ampliar o uso do processo proposto de desenvolvimento de produtos para mais produtos. Ou seja, escalar o processo para toda a organização.
- Realizar análise de tempo das atividades do processo;

## REFERÊNCIAS BIBLIOGRÁFICAS

BENEDICT, T., BILODEAU, N., VITKUS, P., POWELL, E., MORRIS, D., SCARSIG, M., LEE, D., FIELD, G., LOHR, T., SAXENA, R., *et al.* **BPM CBOOK Version 3.0: Guide to the Business Process Management Common Body Of Knowledge version 3.0.** 3rd Edition, CreateSpace Independent Publishing Platform, 2013.

BRAGA, F. A. M. **Um panorama sobre o uso de práticas DevOps nas indústrias de software.** 2015. 123f. Dissertação (Mestrado) – Universidade Federal de Pernambuco.

CANKURTARAN, P., LANGERAK, F., GRIFFIN, A. Consequence of New Product Development Speed: A Meta-Analysis. **Journal of Product Innovation Management**, Vol. 30 No. 3, 2013, pp. 465-486.

CARVALHO, M., PALADINI, E. **Gestão da Qualidade: Teoria e Casos.** 2ª Ed. Rio de Janeiro: Elsevier: ABEPRO, 2012.

CERVONE, H. Understand agile project management methods using Scrum. OCLC Systems & Services: **International digital library perspectives**, Vol. 27 No. 1, pp. 18-22, 2011.

DUVALL, P. **Continuous integration: improving software quality and reducing risk.** Pearson Education, Inc. 2007.

DYBA, T., DINGSØYR, T. Empirical studies of agile software development: A systematic review. **Information and Software Technology**, v. 50, pp. 833-859, 2008.

HIGHSMITH, J. **Agile Software Development Ecosystems.** Adilson Wesley, 2002.

JUNIOR, E., BRODBECK, A., COST, C. Análise das relações dos elementos de alinhamento estratégico entre negócio e tecnologia da informação com o processo de desenvolvimento de produto. **RAM, REV. ADM. MACKENZIE**, V. 13, N. 2, pp. 135-170, 2012.

KIM, G., HUMBLE, J., DEBOIS, P., WILLIS, J. **The Devops Handbook – How to create World-Class Agility, Reliability, & Security in Technology Organizations.** First Ed. Portland: IT Revolution Press, LLC, 2016.



KRUCHETEN, P. Contextualizing agile software development. **Journal of software: evolution and process**, v. 25, pp. 351-361, 2013.

MELO, O., SANTOS, V., KATAYAMA, E., CORBUCCI, H., PRIKLANDNICKI, R., GOLDMAN, A., KON, F. The evolution of agile software development in Brazil. **Journal of the Brazilian Computer Society**, v. 19, pp 523-552, 2013.

MENDOZA, V. Y., SILVEIRA, D. S. Verificando a compreensão do BPMN com gestores de negócios. **Revista Brasileira de Computação Aplicada**, v. 9, n. 4, pp. 60-75, 2017.

MORGAN, J., LIKER, K. **The Toyota product development system: integrating people, process and technology**. New York: Productivity Press, 2006.

PRESSMAN, R. S. **Engenharia de Software: Uma abordagem profissional**. 7. ed. Porto Alegre: AMGH, 2011.

ROZENFELD, H., FORCELLINI, F. A., AMARAL, D. C., TOLEDO, J. C., SILVA, S. L.; ALLIPRANDINI, D. H.; SCALICE, R. K. **Gestão de Desenvolvimento de Produtos: Uma referência para a melhoria do processo**. São Paulo: Saraiva, 2006.

SOUMANLAINEN, T., SALO, O., ABRAHAMSSON, P., SIMILÄ, J. Software product roadmapping in a volatile business environment. **The Journal of System and Software**, V. 84, N. 6, pp. 958-975, 2011.

SHAHIN, M., BABAR, M., ZHU, L. Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices. **IEEE Access**, 5, pp. 3909-3943, 2017.

STAHL, D., BOSCH, J. Cinders: The continuous integration and delivery architecture framework. **Informations and Software Technology**, 83, pp. 76-93, 2017.

TYAGI, S., CHOUDHARY, A., CAI, X., YANG, K. Value stream mapping to reduce the lead-time of a product development process. **International Journal of Production Economics**, V. 160, pp. 202-212, 2015.

# ANEXO I

## PROCESSO DESENVOLVIMENTO DE PRODUTO DE SOFTWARE

