

**UNIVERSIDADE EVANGÉLICA DE GOIÁS – UniEVANGÉLICA  
BACHARELADO EM ENGENHARIA DE SOFTWARE**

**GUSTAVO DOS SANTOS SOBRINHO  
RAPHAELA PAOLLA DOS SANTOS**

**UTILIZAÇÃO DE DEVOPS EM UMA FÁBRICA DE SOFTWARE ACADÊMICA**

**ANÁPOLIS - GO  
2021-01**

**GUSTAVO DOS SANTOS SOBRINHO  
RAPHAELA PAOLLA DOS SANTOS**

**UTILIZAÇÃO DE DEVOPS EM UMA FÁBRICA DE SOFTWARE ACADÊMICA**

Trabalho de Conclusão de Curso I apresentado como requisito parcial para a conclusão da disciplina de Trabalho de Conclusão de Curso I do curso de Bacharelado em Engenharia de Software do Universidade Evangélica de Goiás – UniEVANGÉLICA.

Orientador(a): Prof. Millys Fabrielle Araujo Carvalhaes.

**ANÁPOLIS - GO  
2021-01**

**GUSTAVO DOS SANTOS SOBRINHO  
RAPHAELA PAOLLA DOS SANTOS**

**UTILIZAÇÃO DE DEVOPS EM UMA FÁBRICA DE SOFTWARE ACADÊMICA**

Trabalho de Conclusão de Curso I apresentado como requisito parcial para a conclusão da disciplina de Trabalho de Conclusão de Curso I do curso de Bacharelado em Engenharia de Software do Universidade Evangélica de Goiás – UniEVANGÉLICA.

Orientador(a): Prof. Millys Fabrielle Araujo Carvalhaes

Aprovado(a) pela banca examinadora em [11] de [Junho] de 2021, composta por:

---

Prof. Millys Fabrielle Araujo Carvalhaes  
Orientador

---

Prof. Raissa dos Santos Vieira

## Resumo

A Indústria de Software atual possui a necessidade de entregar funcionalidades e melhorias de seus sistemas cada vez mais rápido e com maior qualidade, este fato se intensifica quando olhamos o contexto de uma fábrica de software acadêmica pois o produto desenvolvido passa por entregas frequentes, tendo em vista que a melhoria do sistema caminha em paralelo com a maturidade adquirida pela equipe. Assim, objetivou-se estruturar um modelo de processo DevOps, utilizando um ambiente acadêmico de desenvolvimento de software, tratando a integração das equipes e melhoria na cadência de entregas. Através da análise dos processos de desenvolvimento vigentes e utilizando as práticas de DevOps que tratam principalmente a comunicação, colaboração, automação e monitoramento será estruturado um processo que crie um ambiente integrado entre as diversas equipes que participam do desenvolvimento e melhoria do sistema.

**Palavras chave:** Indústria de Software, qualidade do produto, DevOps, desenvolvimento de software, integração.

## Lista de Ilustrações

Figura 1: Continuous Integration.....	13
Figura 2: Continuous Deployment .....	13
Figura 3: Ferramentas para práticas DevOps .....	14
Figura 4: Abrangência do DevOps .....	15

## Lista de abreviaturas e siglas

FS	Fábrica de <i>Software</i>
FSA	Fábrica de <i>Software</i> Acadêmica
FTT	Fábrica de Tecnologias Turing
XP	<i>Extreme Programing</i>
IAC	Infraestrutura como Código
TI	Tecnologia da Informação
E/S	Entrada e Saída

## Sumário

<b>1</b>	<b>PROBLEMA.....</b>	<b>7</b>
<b>2</b>	<b>OBJETIVOS.....</b>	<b>8</b>
<b>2.1</b>	<b>Objetivo Geral .....</b>	<b>8</b>
<b>2.2</b>	<b>Objetivos Específicos .....</b>	<b>8</b>
<b>3</b>	<b>JUSTIFICATIVA.....</b>	<b>9</b>
<b>4</b>	<b>FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>10</b>
<b>4.1</b>	<b>O que é e como funciona uma Fábrica de software? .....</b>	<b>10</b>
4.1.1	Fábrica de software acadêmica .....	10
<b>4.2</b>	<b>Cultura Organizacional.....</b>	<b>11</b>
<b>4.3</b>	<b>Processo de software .....</b>	<b>11</b>
4.3.1	Modelo de Processo Incremental .....	12
<b>4.4</b>	<b>DevOps .....</b>	<b>12</b>
<b>5</b>	<b>METODOLOGIA .....</b>	<b>16</b>
<b>6</b>	<b>CRONOGRAMA .....</b>	<b>18</b>
<b>7</b>	<b>RESULTADOS ALCANÇADOS .....</b>	<b>19</b>
<b>8</b>	<b>RESULTADOS ESPERADOS.....</b>	<b>20</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>21</b>
	<b>ANEXO A - PROCESSO DA EQUIPE DE DOCUMENTAÇÃO DA FTT.....</b>	<b>23</b>
	<b>ANEXO B - PROCESSO DA EQUIPE DE DESENVOLVIMENTO DA FTT .</b>	<b>23</b>
	<b>ANEXO C - PROCESSO DA EQUIPE DE TESTE DA FTT .....</b>	<b>24</b>
	<b>ANEXO D - PROCESSO DA EQUIPE DE BANCO DA FTT .....</b>	<b>24</b>
	<b>ANEXO E - PROCESSO GERAL DA FTT .....</b>	<b>25</b>

## 1 PROBLEMA

A Indústria de Software sofre diversas mudanças com o decorrer do tempo, buscando sempre a melhoria de seus produtos de acordo com a necessidade do seu cliente, tanto no desenvolvimento quanto nas melhorias, visando manter a melhor qualidade nos serviços (GUERRERO, 2020). Em meio a este ambiente, as mesmas notam a necessidade de incrementos para a melhoria de sua produtividade, como o uso de ferramentas e frameworks para dar suporte aos processos contidos no desenvolvimento do software, visando adquirir os resultados esperados com alta qualidade em um ambiente de desenvolvimento contínuo de serviços (VIRMANI, 2015).

Desenvolvedores são cobrados por entregas de funcionalidades em aplicações que geram valor, enquanto a equipe de infraestrutura é cobrada pela estabilidade do sistema, assim ambas equipes necessitam de uma maior interação melhorando a agilidade na implementação e em inovações contínuas. Assim, indústrias com uma melhor comunicação onde colaboradores e líderes sabem como agir e estão mais alinhados com seu papel, objetivos e o funcionamento do processo como um todo alcançam uma maior agilidade na entrega de funcionalidades (CRUZ, 2019).

A ausência dessa integração pode gerar retrabalhos para os desenvolvedores e atrasos para a equipe de operações, fazendo com que a qualidade do produto caia e perca sua agilidade em decorrência da destoante cultura e práticas organizacionais vividas por cada equipe. Assim a aplicação da cultura DevOps procura mudar a mentalidade tanto da equipe de desenvolvimento quanto da equipe de operações unificando o cenário de atuação vivido por ambas, trazendo uma integração que visa melhorar a efetividade do desenvolvimento de funcionalidade e correções do sistema, integrando-as como um objetivo comum a todos (BARBOSA, 2019).

Porém, ainda é uma tarefa desafiadora sua implementação, por conter uma infinidade de informações, práticas e ferramentas relacionadas, de modo que ainda não está claro como essa rica, porém ainda dispersa, quantidade de informação é organizada por não conter um guia ou framework estruturado. Assim, como estruturar e implementar um processo DevOps em uma fábrica de software acadêmica (FSA)?

## **2 OBJETIVOS**

### **2.1 Objetivo Geral**

- Desenvolver um processo DevOps para aplicação na Fábrica de Tecnologias Turing (FTT).

### **2.2 Objetivos Específicos**

- Estudar e entender as práticas DevOps;
- Analisar o processo de desenvolvimento e implementação da FTT;
- Estruturar um processo unificado de desenvolvimento e implementação;
- Adaptar a ferramenta GitLab para a centralização do processo;
- Validar o processo proposto e seus resultados.

### 3 JUSTIFICATIVA

Uma das prioridades em uma Fábrica de Software (FS), são as entregas do produto. No decorrer deste processo alguns impedimentos podem ocorrer, como informações perdidas, atrasos nos testes finais ou até uma má priorização do backlog. Grandes projetos na área de Tecnologia da Informação (TI) ultrapassam em média 45% do orçamento original e 7% do cronograma original, sendo que no geral mais de 85% dos projetos de software atrasam (BARBOSA, 2017).

A cultura DevOps descreve abordagens que ajudam a acelerar os processos necessários para levar uma ideia do planejamento à implantação em um ambiente de desenvolvimento no qual ela seja capaz de gerar valor para o usuário dentro do prazo estimado. Essas ideias podem ser um novo recurso de software, uma solicitação de correção de bug. Essas abordagens exigem comunicação frequente entre as equipes, trabalho colaborativo e empatia com os demais membros das equipes, assim o tempo para entendimento do objetivo do trabalho é reduzido, a taxa de erros diminui e é possível validar e corrigir possíveis problemas de forma mais efetiva (KIM et. al, 2018). Ao final de cada iteração, o software se torna mais completo, com novas funcionalidades e menos bugs. (SATO, 2014)

Sua importância dentro das equipes de desenvolvimento e operações vem para permitir ciclos mais curtos, com a ampliação da sua capacidade de inovar e ainda realizar uma entrega com uma maior frequência. Permitindo assim uma entrega mais rápida do que as abordagens tradicionais (AWS, 2020). A integração de equipes com um foco no desempenho e comunicação, pode gerar uma maior confiança entre as equipes e maior liberdade entre os profissionais. Este procura integrar todo o processo de desenvolvimento de software, englobando a análise, desenvolvimento e operações (HUMBLE & FARLEY, 2010).

Segundo pesquisa da Forrester realizada com 230 empresas em 2015, mais de 40% das companhias pesquisadas estão executando programas de Agile combinado com DevOps. Desse total, 83% obtiveram mais eficiência na entrega. Os dados evidenciam também que 88% conseguiram fazer lançamentos mais frequentes e alcançaram retorno comercial mais rápido (ANDREAZZA, 2018).

A inserção da cultura DevOps em uma fábrica acadêmica de desenvolvimento de software visa utilizar as práticas da cultura para tornar o ambiente mais integrado, permitindo assim que a informação e conhecimento fluam melhor entre os membros das equipes e que o software possa evoluir em conjuntos com o conhecimento dos alunos, de forma incremental e estruturada, evitando atrasos no produto final.

## 4 FUNDAMENTAÇÃO TEÓRICA

### 4.1 O que é e como funciona uma Fábrica de software?

O termo FS remete a utilização de conceitos da indústria geral em um ambiente de produção. Um processo formado por passos de subprocessos parcialmente ordenado, definindo a ideia de aplicar conceitos da indústria em geral em ambientes de desenvolvimento de *software*, de forma a aumentar a produtividade e diminuir prazos e custos, tornando o processo mais independente do fator humano. (AGUILAR, 2011)

Uma FS realiza a criação de produtos de acordo com a necessidade do seu cliente, fazendo uso de processos e operações que façam garantir a qualidade, adequando a um ciclo de desenvolvimento e produtividade (FERNANDES, 2004).

Para um bom funcionamento, nota-se a necessidade de dividir a fábrica em setores: atendimento de clientes, negociação e especificação de necessidades da área usuária, planejamento e controle de produção, definição de objetos a serem utilizados ou desenvolvidos e a equipe de garantia que verifica se o produto final atende a todas as especificações. Cada processo tem como foco um cenário diferente de produção e produto, sendo assim prioriza diferentes artefatos com base em sua metodologia. Cada etapa, da metodologia adotada, devolve parte do desenvolvimento do software em si, desde sua concepção inicial até o produto final, sendo cada etapa continuação da etapa anterior (AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013).

#### 4.1.1 FÁBRICA DE SOFTWARE ACADÊMICA

Uma FSA é uma modalidade dentro da FS composta pelo corpo acadêmico das instituições. Durante o período de graduação estimula-se a interdisciplinaridade através da simulação do ambiente empresarial, diferenciais alcançados na busca por novas metodologias e processos para o desenvolvimento de um software. Provocando o corpo docente a estar sempre atualizado e capacitado, por causa das constantes mudanças no ambiente criado pela FSA (OLIVEIRA; NETO, 2003).

Estimulando o empreendedorismo e a pesquisa na FSA, pode ser criado um espaço de tecnologias que possam ser idealizadas e desenvolvidas por alunos. Proporcionando um ambiente de aprendizado para os alunos, onde possa ser compartilhado ideias e estudos,

realizando entregas de produtos reais através de um processo definido. (ROMANHA; MUNIZ JR; DALE LUCHE, *apud* IZURIETA, 2019).

## **4.2 Cultura Organizacional**

O termo cultura organizacional foi estabelecido no final do século 18. É uma abreviatura para a gestão do pensamento e comportamento humanos. Só foi efetivamente usado um século depois. É uma forma de comparar o comportamento humano de uma forma padronizada, também usando um método prescritivo para estudar a cultura organizacional, que é um representante típico da teoria do desenvolvimento organizacional, que regula e prescreve a cultura que levará a organização ao sucesso (RUSSO, 2010).

Armbrust (2021) define que cultura organizacional é um conjunto de elementos (crenças, valores e normas) que influenciam o clima de uma empresa, ela se destaca como recurso estratégico para a busca da competitividade. Portanto, é muito importante para você definir como a empresa se posiciona e como se destaca no mercado.

Russo (2010) então diz que a cultura organizacional pode ser definida como um sistema desenvolvido e profundamente enraizado de valores que estimula os membros da organização, quando adequadamente gerenciado, a produzirem mais esforço, mais criatividade e melhor desempenho organizacional. A cultura organizacional representada pelos valores compartilhados entre os membros da organização provê uma identidade.

## **4.3 Processo de software**

Um processo de software pode ser entendido como uma sequência estruturada de atividades a serem realizadas durante sua produção. Um processo é um conjunto de etapas que envolvem atividades, restrições e recursos para alcançar uma saída desejada. Sendo assim, é visto como atividades e resultados que quando interligados, geram um produto de software. (SOMMERVILLE, 2004).

Segundo Pressman (2011), um processo genérico estabelece cinco atividades metodológicas: comunicação, planejamento, modelagem, construção e entrega. Além disso, há mais um conjunto de atividades de apoio às atividades guarda-chuva que podem ser utilizadas, elas são aplicadas ao longo do projeto e ajudam a administrar riscos e garantir a qualidade. Um fluxo de processo descreve como são organizadas essas cinco atividades metodológicas.

### 4.3.1 MODELO DE PROCESSO INCREMENTAL

Na visão de Pressman (2011) O modelo incremental de desenvolvimento de software mistura elementos dos fluxos de processo lineares e paralelos. ele aplica uma sequência linear de produção, de forma escalonada, sendo assim à medida que o tempo avança ele gera um novo incremento como em um fluxo de processos evolucionários.

O processo incremental de desenvolvimento pode ser visto como a ideia de “aumentar um pouco” o âmbito do sistema. Uma mansão que foi construída a partir de uma casa com apenas algumas divisões utilizando incrementos cíclicos devidamente priorizados, é um bom exemplo de como o processo em questão funciona Ramos (2006). Para que esse modelo de desenvolvimento seja viável é preciso que a equipe responsável possua suas cinco atividades metodológicas bem estruturadas, com foco principal na comunicação e planejamento (RAMOS, 2006).

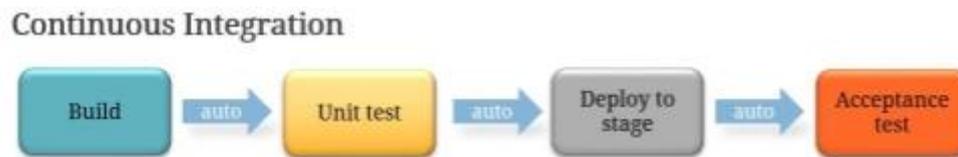
## 4.4 DevOps

O DevOps é frequentemente referido pelos profissionais da área das tecnologias da informação como um movimento cultural ou profissional, que apresenta uma nova abordagem de entrega de aplicações informáticas (*software*), através da colaboração entre as equipes de desenvolvimento e das equipes de operações. Tem subjacente um conjunto diversificado de princípios, relacionados com cultura, automatização, lean, monitorização e partilha; e práticas, tais como, *continuous integration* (Figura 1) e *continuous deployment* (Figura 2) (SOUSA, 2019).

- De acordo com SATO (2014) *continuous integration* (Integração contínua) é uma das práticas originais de Extreme Programming (XP), incentivando desenvolvedores a frequentemente integrar seu trabalho para encontrar e resolver possíveis problemas rapidamente. Depois que o desenvolvedor conclui uma tarefa que pode compartilhar o código com a equipe, seguindo um processo disciplinado para garantir suas mudanças, não adicionando problemas, podendo integrar com o resto do código.
- *Continuous deployment* (Implementação contínua) para SATO (2014) é uma prática de colocar cada commit em produção, o que geralmente significa várias implantações em produção todos os dias.

As imagens a seguir representam parte do ciclo de vida da integração (Figura 1) e implementação (Figura 2) contínua do DevOps:

Figura 1: *Continuous Integration*



Fonte: Adaptado de MATSUMOTA, 2018

Figura 2: *Continuous Deployment*



Fonte: Adaptado de MATSUMOTA, 2018

SACOLICK (2018) complementa dizendo que as práticas de DevOps também incluem:

- Controle de versão e estratégias de ramificação.
- *Containers* que padronizam e isolam os ambientes de tempo de execução do aplicativo.
- *Infrastructure as Code* (IAC), que permite criar scripts na camada de infraestrutura.
- Monitorando os pipelines Devops e a integridade dos aplicativos em execução.

Controle de versão, de acordo com Shapiro (2020) são uma categoria de ferramentas de software que ajudam equipes de software a gerenciar alterações no código-fonte com o passar do tempo. *Containers*, também conhecidos como containers linux, de acordo com PRADO (2019) é uma instância da camada de usuário do Linux, que aloca muitos recursos (CPU, memória, E/S) e é executada de forma isolada do resto do sistema. Um contêiner pode executar apenas um aplicativo ou todo o sistema de arquivos (rootfs), para que vários contêineres possam ser iniciados e executados ao mesmo tempo. IAC é um método de automação de infraestrutura de TI, utilizado principalmente para permitir que equipes de DevOps gerenciem e configurem



Figura 4: Abrangência do DevOps



Fonte: Adaptado de BARBOSA, 2019

É um fenômeno em expansão, mas cuja adoção ao nível das organizações ainda está numa fase embrionária, necessitando de mais investigação no sentido de clarificar os benefícios, custos e barreiras à adoção do mesmo.

Foi com este objetivo subjacente que se conduziu o trabalho de investigação apresentado neste relatório, um estudo de caso de uma organização de grande dimensão da área das telecomunicações, que está atualmente a passar por um processo de migração para a cultura DevOps. Através do caso de estudo foi possível verificar o “antes” e “depois” da adoção de DevOps na organização, e estudar a sua adoção nas diferentes vertentes.

Como principais resultados identificam-se a revisão da literatura efetuada e a caracterização vários aspetos como, por exemplo, as práticas (que os entrevistados identificaram como mais relevantes a *continuous integration* e *continuous deployment*), os benefícios, barreiras, (ex: resistência à mudança). De destacar, ainda, como principal fator influenciador do sucesso na adoção/implementação de DevOps, à semelhança de outras iniciativas nas organizações, o apoio da gestão de topo (SOUSA, 2019).

## 5 METODOLOGIA

Inicialmente será realizado um estudo das práticas contidas em um processo DevOps, dedicando-se à compreensão de seus pontos de contribuição na comunicação, colaboração, automação e monitoramento em prol da melhoria da qualidade no desenvolvimento, mas também suas dificuldades de aplicação e estruturação para serem colocadas em vigor. Em paralelo serão realizadas pesquisas bibliográficas que tratam da criação e estruturação de processos ágeis, tendo como objetivo consolidar uma base de conhecimento mais amplo sobre os processos de desenvolvimento de software.

Assim, a segunda etapa do projeto consistirá na análise dos processos de desenvolvimento e operações dentro da FTT, assim como a interação entre as respectivas equipes, juntamente com uma pesquisa de conhecimento voltada para o processo híbrido utilizado. Com um olhar crítico às possíveis causas e efeitos já identificados, será feita uma comparação dos processos legados e vigentes da FTT, permitindo realizar uma estrutura de implementação do processo DevOps baseado nos objetivos e estrutura de trabalho dela. Tais informações serão levantadas através da observação participante e questionários para consolidação de estatísticas ali encontradas.

A implementação do processo DevOps será integrada aos atuais processos com as devidas adaptações necessárias na rotina, cultura e metodologias empresariais, reduzindo o impacto no cotidiano das equipes tornando a implementação mais fluida, visando atender de forma responsiva e adaptável. Utilizando como base o modelo proposto por Dias (2018), gerente de cloud da Mandic Cloud Solutions, as etapas de implementação de DevOps serão as seguintes:

- 1 - Realizar mudanças na cultura dentro do ambiente de trabalho para uma equipe integrada, uma vez que o DevOps visa uma maior integração entre os membros e as equipes. Trabalhar este ponto na cultura organizacional faz com que a interação entre as equipes não seja uma tarefa a ser feita e sim uma rotina de trabalho.
- 2 - Analisar os impactos da mudança e se as metodologias ágeis dentro da FS, contém a mesma visão para planejar atividades focadas em resultados, automação e integração.

3 - Automatizações, a parte técnica suporta a gestão de pessoas e vice-versa. Nessa parte será utilizada a ferramenta GitLab para integrar ferramentas e processos, criando um ambiente onde os artefatos gerados pelas equipes possam se conectar de uma maneira mais simples e fluida.

4 – Automatizar alguns processos de teste. Antes de realizar os testes automatizados, é indispensável fazer testes de regressão, aceitação, integração, entre outros, para garantir que o ciclo automatizado criado está adequado.

5 - Utilizar métricas e objetivos para especificar e planejar o que será utilizado durante o projeto.

6 - Realizar o mapeamento da evolução dos projetos.

Passando pelas etapas, serão observadas as respostas das equipes quanto a produtividade e a qualidade dentro das sprints criadas, validando assim cada uma das 6 etapas de implantação da cultura DevOps, podendo haver novas melhorias e adaptações caso seja observada necessidade.



## 7 RESULTADOS ALCANÇADOS

Os resultados obtidos a partir da realização da primeira etapa, através de pesquisas mostrando como uma boa integração dentro de uma empresa faz com que o processo ágil seja ainda mais eficaz, compreendendo suas dificuldades de implementação e a qualidade que o mesmo proporciona, tanto na melhoria do produto como na melhoria dos processos da FS. Baseando-se a priori no fato de que empresas implementam e entregam soluções de software 30 vezes mais rápido e reduzem até 200 vezes os prazos de entrega inicialmente idealizados quando possuem uma integração estruturada.

Foram realizadas análises nos processos das equipes de documentação, desenvolvimento, teste, banco e no processo geral da FTT para compreender os pontos a serem integrados às novas implementações e estruturação das operações. Juntamente foram realizadas buscas a respeito da adaptação do DevOps e metodologias ágeis com as fases de elaboração do produto. Sendo assim pode-se criar uma boa base de conhecimento para o planejamento e estruturação de mudanças nos processos e uma melhor visão de como integrá-los.

Foi possível também mapear as melhorias implementadas ao longo do tempo em cada equipe e entender melhor a evolução e maturidade em cada etapa do desenvolvimento de sistemas. Este conhecimento nos permitiu ter uma visão mais abrangente do ambiente da FTT, gerando assim maior clareza nos objetivos e temas a serem abordados. Um ponto importante é a diferença na maturidade dos processos, algumas equipes tem um processo mais antigo que passou por melhorias e adaptações através de estudos fundamentados ao longo do tempo, outras tem seu processo relativamente novo, sendo assim, é importante entender as mudanças feitas e o que as ocasionou para que o resultado esperado venha para complementar o ambiente sem inviabilizar estudo prévios, ao passo que também é importante entender as melhorias necessárias nos processos mais novos, onde o tema proposto neste trabalho pode vir de forma a complementar e melhorar o processo das equipes em questão.

## **8 RESULTADOS ESPERADOS**

A principal proposta deste trabalho é não somente melhorar o processo de desenvolvimento, mas mostrar seus resultados práticos, assim espera-se propor mudanças efetivas na rotina e processos da FTT, melhorando não apenas a qualidade dos produtos desenvolvidos, mas também a manutenção e evolução dos mesmos, realizando as atividades com maior comunicação e melhor interação entre as equipes evitando dualidade de informações.

Além disso, espera-se a obtenção da experiência que será gerada ao vivenciar a aplicação de conceitos de diferentes metodologias em um ambiente real de uma FSA e com base nos conhecimentos adquiridos, realizar melhorias no processo de desenvolvimento. Também é esperado que este trabalho possa incentivar outros membros a produzirem pesquisas semelhantes ou melhorias para o processo implementado.

Não menos importante, espera-se que este trabalho gere um crescimento profissional para seus autores através dos estudos que serão realizados durante o projeto, tanto através da revisão de literatura quanto através das pesquisas e aplicações realizadas em campo.

## REFERÊNCIAS BIBLIOGRÁFICAS

- AGUILAR, E. **O que é e como funciona uma fábrica de software**. Profissionais TI, 2011. Disponível em: <<https://www.profissionaisiti.com.br/o-que-e-e-como-funciona-uma-fabrica-de-software/>> Acesso em: 15 de abril de 2021.
- AMARAL, A. **O que é infraestrutura como código?**. Tivit, 2018. Disponível em: <<https://blog.tivit.com/o-que-e-infraestrutura-como-codigo>> Acesso em: 10 de maio de 2021.
- ANDREAZZA, M. **Agile e DevOps: união indispensável para a TI**. Baguete, 2018. Disponível em: <<https://www.baguete.com.br/noticias/14/08/2018/agile-e-devops-uniao-indispensavel-para-a-ti>> Acesso em: 15 de abril de 2021.
- ARMBRUST, G. **Cultura organizacional: o que é, importância, tipos e exemplos**. Gupy, 2021. Disponível em: <<https://www.gupy.io/blog/cultura-organizacional>> Acesso em: 10 de maio de 2021.
- BARBOSA, D. **Importância de DevOps para as organizações**. CEDRO, 2019. Disponível em: <<https://blog.cedrotech.com/importancia-de-devops-para-as-organizacoes/>> Acesso em: 06 de maio de 2021.
- BARBOSA, M. **Por que mais de 85% dos projetos de software atrasam?**. McKinsey & Company, [s. l.], n. October 2012, p. 1–11, 2017. Disponível em: <<https://inforchannel.com.br/2017/11/08/por-que-mais-de-85-dos-projetos-de-software-atrasam/>> Acesso em: 25 de abril de 2021.
- CRUZ, D. S. **3 problemas que uma cultura organizacional bem definida pode resolver em sua empresa**. Santo Caos, 2019. Disponível em: <<https://www.santocaos.com.br/post/3-problemas-que-uma-cultura-organizacional-bem-definida-pode-resolver-em-sua-empresa>> Acesso em: 21 de abril de 2021.
- DIAS, G. **Adoção da cultura DevOps: O que ainda falta?**. Baguete, 2018. Disponível em: <<https://www.baguete.com.br/noticias/08/08/2018/adocao-da-cultura-devops-o-que-ainda-falta.>> Acesso em: 15 de abril de 2021.
- FORSGREN, N. et al. **2015 State of DevOps Report**. PuppetLabs. 2015.
- GUERRERO, J. **Tendencias en DevOps: un mapeo sistemático de la literatura**. Universidad del Cauca, 2020. Disponível em: <[https://www.researchgate.net/profile/Cesar-Calvache/publication/344478031\\_Tendencias\\_en\\_DevOps\\_un\\_mapeo\\_sistemico\\_de\\_la\\_literatura/links/5f7b493aa6fdcc0086576b2b/Tendencias-en-DevOps-un-mapeo-sistemico-de-la-literatura.pdf](https://www.researchgate.net/profile/Cesar-Calvache/publication/344478031_Tendencias_en_DevOps_un_mapeo_sistemico_de_la_literatura/links/5f7b493aa6fdcc0086576b2b/Tendencias-en-DevOps-un-mapeo-sistemico-de-la-literatura.pdf)> Acesso em: 14 de abril de 2021.
- HUMBLE, J., & FARLEY, D. **Continuous Delivery**. Pearson Education. 2010.
- LEANDRO, Peterson et al. **Dificuldades encontradas durante a implantação de sistema de custos: um estudo realizado com base em artigos do congresso brasileiro de custos**. [s. l.]. In: Congresso Brasileiro de Custos, 19., 2012. Disponível em: <<https://anaiscbc.emnuvens.com.br/anais/article/download/328/328>> Acesso em: 14 de abril de 2021.

MARQUES, J. R. **Quais são os principais problemas organizacionais**. Instituto Brasileiro de Coaching, 2019. Disponível em: <<https://www.ibccoaching.com.br/portal/rh-gestao-pessoas/quais-sao-principais-problemas-organizacionais/>> Acesso em: 15 de abril de 2021.

MATSUMOTA, L. **Cultura DevOps na sua empresa**. Digital Strategy and IT Innovation, 2018. Disponível em: <<https://leonardo-matsumota.com/2018/10/14/cultura-devops-na-sua-empresa/>>

MOREIRA, D. **A Importância da Cultura Organizacional**. Incomum, 2019. Disponível em: <<https://incomum.in/blog/desvantagem-de-nao-ter-cultura-organizacional>> Acesso em: 15 de abril de 2021.

PRADO, S. **O que é um container linux?**. Sergio Prado Org, 2019. Disponível em: <https://sergioprado.org/o-que-e-um-container-linux/>

PRATT, M. K. **Cinco obstáculos à adoção do DevOps**. EUA, 2017. Disponível em: <<https://cio.com.br/tendencias/cinco-obstaculos-a-adocao-do-devops/>> Acesso em: 16 de abril de 2021.

PRESSMAN, R. **Engenharia de Software: Uma Abordagem Profissional**. 7ª Ed., São Paulo: McGraw-Hill, 2011.

RAMOS, R. **Treinamento Prático em UML**. São Paulo: Digerati Books, 2006.

RUSSO, G. M. **Diagnóstico da Cultura Organizacional**. Rio de Janeiro: Elsevier Editora Ltda., 2010.

SACOLICK, I. **Práticas recomendadas de Devops: 5 métodos que você deve adotar**. CIO, 218. Disponível em: <<https://cio.com.br/gestao/praticas-recomendadas-de-devops-5-metodos-que-voce-deve-adotar/>> Acesso em: 15 de abril de 2021.

SATO, D. **DevOps na prática: entrega de software confiável e automatizada**. Casa do Código, 2014.

SOUSA, L. **DevOps - Estudo de Caso**. ISCAC, 2019. Disponível em: <<https://comum.rcaap.pt/handle/10400.26/31932>>

SOMMERVILLE, I. **Software engineering**. 7th. ed. Addison-Wesley, 2004.

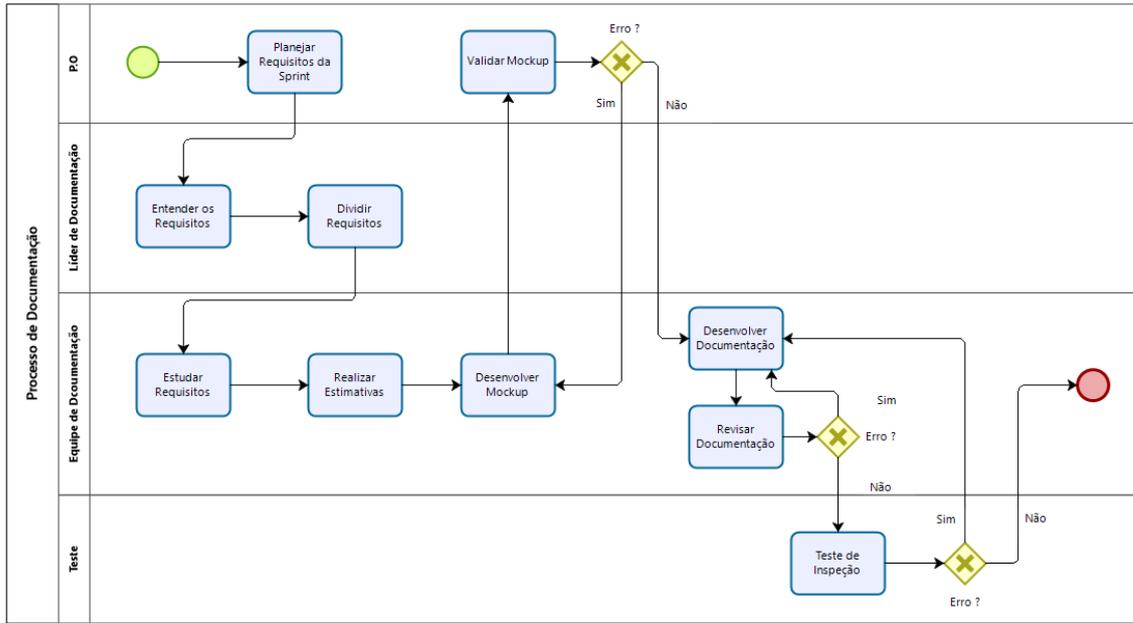
SHAPIRO, J. **O que é controle de versão**. Atlassian, 2020. Disponível em: <<https://www.atlassian.com/br/git/tutorials/what-is-version-control#benefits-of-version-control>>

VARAJÃO, J. **The many facets of information systems**. International Journal of Information Systems and Project Management. 2018.

VIRMANI, M. **Understanding Devops & Bridging The Gap From Continuous Integration To Continuous Delivery**. Fifth International Conference on the Innovative Computing Technology (INTECH 2015). IEEE, 78–82. New York, NY, USA: IEEE.

## ANEXO A - PROCESSO DA EQUIPE DE DOCUMENTAÇÃO DA FTT

Figura 5: Processo da equipe de documentação da FTT

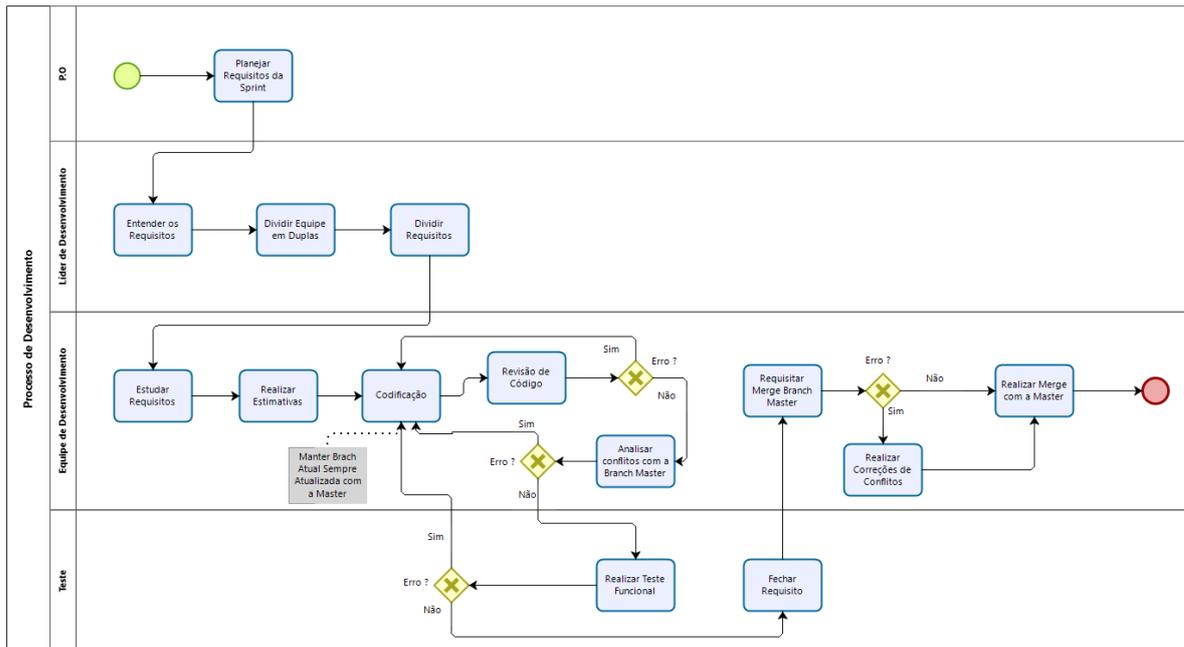


Powered by bizagi Modeler

Fonte: Disponibilizado pela FTT

## ANEXO B - PROCESSO DA EQUIPE DE DESENVOLVIMENTO DA FTT

Figura 6: Processo da equipe de desenvolvimento da FTT

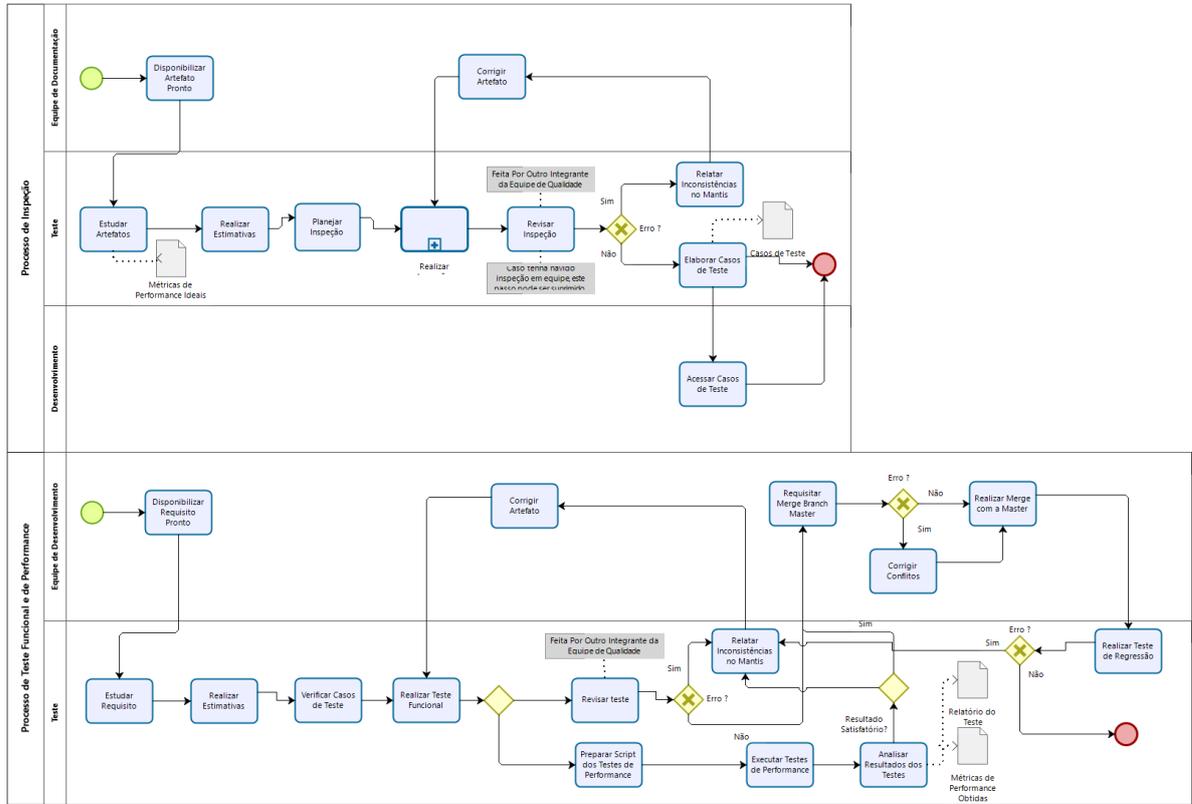


Powered by bizagi Modeler

Fonte: Disponibilizado pela FTT

## ANEXO C - PROCESSO DA EQUIPE DE TESTE DA FTT

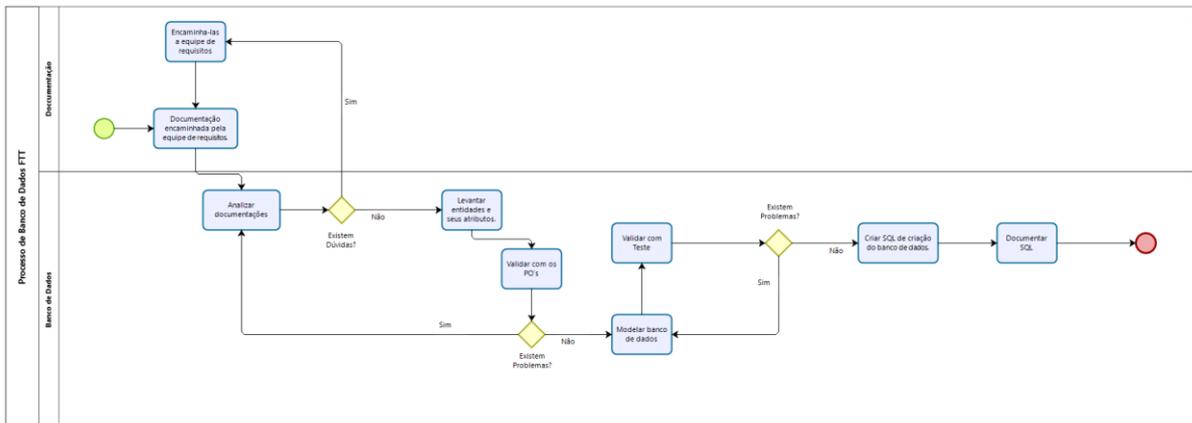
Figura 7: Processo da equipe de teste da FTT



Fonte: Disponibilizado pela FTT

## ANEXO D - PROCESSO DA EQUIPE DE BANCO DA FTT

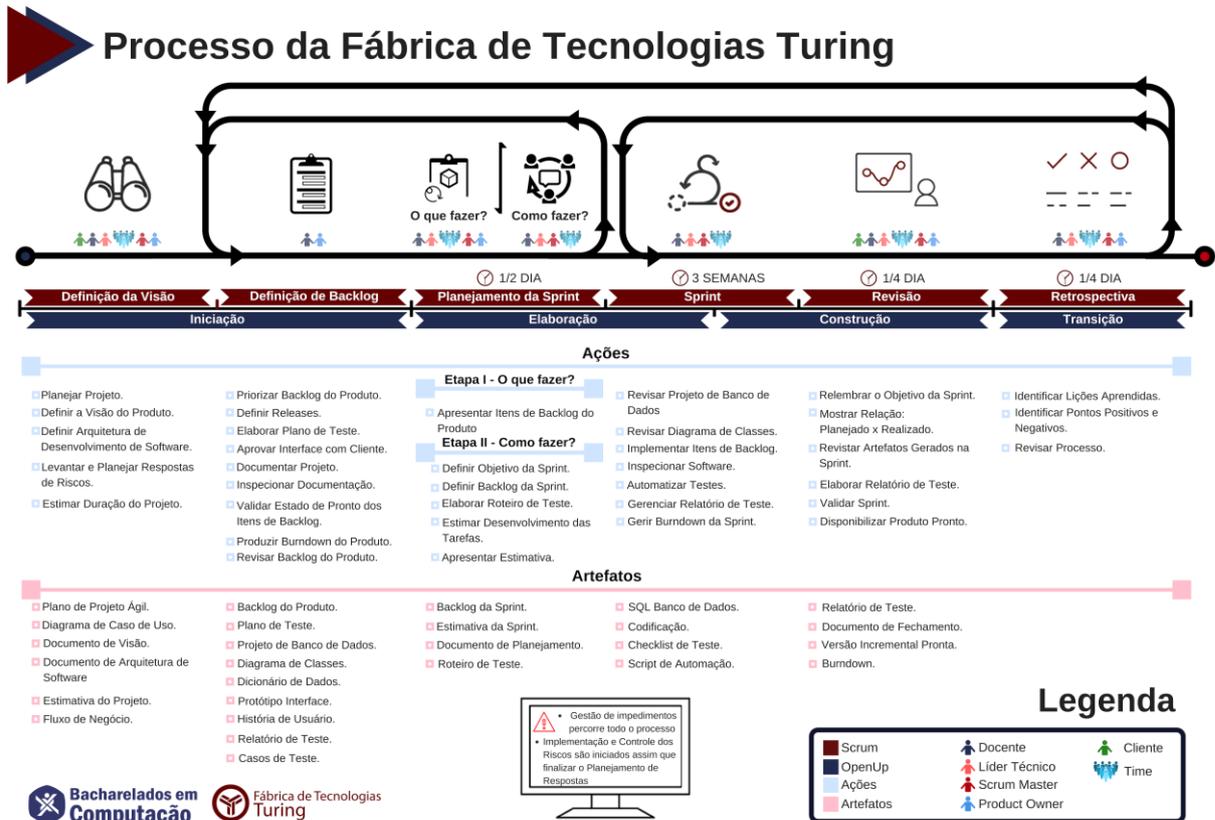
Figura 8: Processo da equipe de banco da FTT



Fonte: Disponibilizado pela FTT

# ANEXO E - PROCESSO GERAL DA FTT

Figura 9: Processo geral da FTT



Fonte: Disponibilizado pela FTT