

INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA
DE MINAS GERAIS - *CAMPUS* AVANÇADO ARCOS
BACHARELADO EM ENGENHARIA MECÂNICA

Gabriel Sander Pereira de Oliveira

**DESENVOLVIMENTO DE UM *SOFTWARE* RPA EM
PYTHON: Aumento da produtividade de um processo logístico
em razão da automação de processos e da tomada de decisão
orientada por dados**

Arcos

9 de Dezembro de 2023

GABRIEL SANDER PEREIRA DE OLIVEIRA

DESENVOLVIMENTO DE UM *SOFTWARE* RPA EM PYTHON: Aumento da produtividade de um processo logístico em razão da automação de processos e da tomada de decisão orientada por dados

Trabalho de conclusão de curso apresentado ao curso de Bacharelado em Engenharia Mecânica do Instituto Federal de Minas Gerais *Campus* Arcos, como requisito parcial à obtenção do grau de Bacharel em Engenharia Mecânica.
Orientador: Marcos Paulo Gonçalves Pedroso.

Arcos
9 de Dezembro de 2023

Catálogo na Fonte Biblioteca IFMG - Campus Arcos

O48d Oliveira, Gabriel Sander Pereira de.

Desenvolvimento de um software RPA em Python: aumento da produtividade de um processo logístico em razão da automação de processos e da tomada de decisão orientada por dados. / Gabriel Sander Pereira de Oliveira. - Arcos, 2023.

35 f. : il. color.

Orientador: Marcos Paulo Gonçalves Pedroso.

Trabalho de Conclusão de Curso (Graduação em Engenharia Mecânica) - Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - *Campus Arcos*.

1. Python – linguagem de programação. I. Pedroso, Marcos Paulo Gonçalves (orientador). IV. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - *Campus Arcos*. V. Título.

CDD 005.13

Catálogo: Meriely Ferreira de Almeida - CRB-6/2760



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS GERAIS

Campus Avançado Arcos

Diretoria de Ensino

Docentes Área Técnica

Av. Juscelino Kubitschek, 485 - Bairro Brasília - CEP 35588000 - Arcos - MG
3733515173 - www.ifmg.edu.br

ATA DE DEFESA DO TCC

Aos nove dias do mês de dezembro de dois mil e vinte e três, às 08:00, se reuniu a banca composta pelo **Prof. Dr. Marcos Paulo Gonçalves Pedroso** (orientador), Instituto Federal de Minas Gerais – Campus Avançado Arcos; **Prof. Me. Maurício Lourenço Jorge**, Instituto Federal de Minas Gerais – Campus Avançado Arcos; **Prof. Dr. Reginaldo Gonçalves Leão Junior**, Instituto Federal de Minas Gerais – Campus Avançado Arcos; para avaliar o trabalho intitulado “**DESENVOLVIMENTO DE UM SOFTWARE RPA EM PYTHON: Aumento da produtividade de um processo logístico em razão da automação de processos e da tomada de decisão orientada por dados**”, apresentado pelo aluno **Gabriel Sander Pereira de Oliveira**, como parte dos requisitos para obtenção do Título de Engenheiro Mecânico. Após apresentação e arguição, emitiu-se o parecer “**APROVADO**”, sendo a verificação das modificações sugeridas de responsabilidade do orientador. A banca avaliadora emite, em consenso, o conceito final 75. Nada mais havendo a tratar a defesa foi encerrada às nove e meia e eu, Marcos Paulo Gonçalves Pedroso, lavrei a presente ata que, após lida e aprovada, foi assinada por todos os avaliadores.

Arcos, 09 de dezembro de 2023.



Documento assinado eletronicamente por **Mauricio Lourenco Jorge, Professor**, em 11/12/2023, às 10:50, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Marcos Paulo Gonçalves Pedroso, Professor**, em 20/12/2023, às 10:26, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Reginaldo Goncalves Leao Junior, Professor**, em 20/12/2023, às 16:40, conforme Decreto nº 10.543, de 13 de novembro de 2020.



A autenticidade do documento pode ser conferida no site <https://sei.ifmg.edu.br/consultadocs> informando o código verificador **1766845** e o código CRC **E92C5E61**.

23808.001004/2023-10

1766845v1

AGRADECIMENTOS

De todos os obrigados que eu deveria dar, gostaria de deixar um agradecimento especial para o meu então superior direto, Felipe Coelho. Essa pessoa especial representa uma fase muito importante em minha vida, onde pude enfrentar diversos medos, errar muito, e aprender ainda mais com eles. Sua amizade e bom humor foram importantíssimos para que me sentisse incluído em um ambiente tão hostil. Agradeço por ter acreditado em mim quando nem mesmo eu acreditava.

RESUMO

Em um mundo cada vez mais dinâmico, buscam-se a todo momento maneiras de se realizar tarefas de maneira mais rápida e eficiente; O presente trabalho apresenta uma solução de automação feita em *Python*, que tem como intuito aumentar a produtividade de um processo específico, em uma empresa de distribuição de bebidas, assim como utilizar o princípio de tomada de decisão orientada por dados, para que sejam geradas ações após a análise dos dados gerados. Após o desenvolvimento e implementação da ferramenta, além da notável melhora do ambiente de trabalho devido à redução de trabalhos repetitivos antes realizados por humanos, verificaram-se também melhoras de até 45% em dois indicadores-chave analisados pela empresa.

Palavras-chave: RPA. Automação. Python. Selenium. Produtividade.

ABSTRACT

In an increasingly lively world, people are always looking for ways to perform tasks faster and more efficiently. This work presents an automation solution made in python, which aims to increase the productivity of a specific process, in a beverage distribution company, as well as to use the principle of data-driven decision-making, so that acertive actions are taken after analyzing the generated data. After the development and implementation of the tool, in addition to the notable improvement in the work environment due to the reduction of repetitive work previously performed by humans, there were also improvements of up to 45% in two key indicators analyzed by the company.

Keywords: RPA. Python. Selenium. Productivity.

LISTA DE FIGURAS

Figura 2.1 – Natureza do processo de automação com utilização de robôs	20
Figura 2.2 – Componentes do Selenium	22
Figura 3.1 – Processo de compra e entrega	23
Figura 3.2 – Comunicação de devoluções	24
Figura 3.3 – Processo de tratativa de devoluções	25
Figura 3.4 – Processo de envio de mensagens automatizadas	28
Figura 3.5 – Mensagem padrão	29
Figura 3.6 – Fluxograma atualizado	30
Figura 4.1 – TC pré e pós ferramenta	31
Figura 4.2 – Taxa de reversão x meses do ano	32
Figura 4.3 – Tempo de captação ao longo do dia	33
Figura 4.4 – Tempo de captação x numero de devoluções ao longo do dia	33

SUMÁRIO

1	INTRODUÇÃO	16
1.1	Motivação do trabalho	17
1.2	Justificativa	17
1.3	Objetivos	17
1.3.1	<i>Objetivo Geral</i>	17
1.3.2	<i>Objetivos específicos</i>	18
2	REFERENCIAL TEÓRICO	19
2.1	Produtividade	19
2.1.1	<i>Gestão da produtividade</i>	19
2.2	RPA - <i>Robotic Process Automation</i>	19
2.3	Python	21
2.3.1	<i>Selenium</i>	21
2.4	DDDM - Data driven decision making	22
3	METODOLOGIA	23
3.1	Contextualização e descrição do problema	23
3.1.1	<i>O aplicativo</i>	23
3.1.2	<i>Sobre o processo de entrega</i>	23
3.1.3	<i>Gestão da devolução</i>	24
3.2	Tempo de captação	25
3.3	Reversão	26
3.3.1	<i>Bate-papo com o monitoramento</i>	27
3.4	Desenvolvimento da solução RPA	27

3.5	Subprocessos importantes	29
3.5.1	<i>Armazenar devoluções já tratadas</i>	29
4	RESULTADOS	31
4.0.1	<i>Comparação dos tempos de captação</i>	31
4.0.2	<i>Comparação reversão</i>	31
4.0.3	<i>Escala do monitoramento</i>	32
5	CONCLUSÃO	35
	REFERÊNCIAS	36
	APÊNDICE A – CÓDIGO FONTE	37

1 INTRODUÇÃO

Desde a eclosão da primeira Revolução Industrial, a humanidade testemunhou um progresso sem precedentes. A busca incessante por maneiras mais eficazes de realizar tarefas, em intervalos cada vez mais curtos, com menos esforço e recursos, marcou a era industrial. O século XIX trouxe consigo a revolução digital, multiplicando exponencialmente as opções disponíveis.

Nesse contexto, cresce também a preocupação e atenção a dois termos importantíssimos: eficiência e produtividade. No entanto, formuladores de políticas econômicas e técnicos de diferentes áreas empregam esses termos de maneira equivocada e, até mesmo sem entender seu conceito e técnicas de medição (TUPY; YAMAGUCHI, 1998).

Em se tratando de aumento de produtividade, a RPA (Robotic Process Automation) é uma tecnologia já consolidada no mercado. Ela utiliza ferramentas computacionais para automatizar processos que antes eram inviáveis ou muito custosos. Com a RPA, é possível reduzir significativamente o tempo e os recursos humanos necessários para realizar tarefas repetitivas e rotineiras, liberando os colaboradores para atividades que exigem maior criatividade e análise. (IVANČIĆ et al., 2019). No entanto, os benefícios da utilização deste conceito vão além de produtividade, lucro, velocidade. Em um mundo onde trabalhadores são cada vez mais exigidos, automatizar processos repetitivos pode também significar retirar funcionários de situações insalubres, proporcionando assim um melhor ambiente de trabalho, onde seu tempo e energia poderão ser gastos com tarefas que realmente agregam valor (HOFMANN et al., 2020).

Existem hoje no mercado diversas ferramentas já bem estabelecidas e estáveis, como: *UiPath*, *Automation Anywhere* e *Blue Prism*. *Softwares* de automação oferecem soluções completas, com metodologias claras e interfaces intuitivas. Isso facilita o treinamento de funcionários e aumenta a produtividade em empresas com alta demanda por automação. A disponibilidade de cursos especializados e comunidades ativas contribui para a rápida adaptação dos usuários.

Nesse cenário, existem também diversas linguagens de programação que conhecidamente performam de maneira satisfatória quando o assunto é automação de processos, e são ferramentas imprescindíveis pela busca de resultados mais rápidos e precisos (ABREU et al., 2017). Apesar de não dispor de interfaces amigáveis e intuitivas ao usuário, aqueles que se aventuram no vasto mundo da programação contam com uma maior versatilidade para resolução de problemas, pois já não mais estão presos às funcionalidades de uma

ferramenta específica.

O presente trabalho foi desenvolvido utilizando a linguagem de programação *Python*, por se tratar de uma ferramenta livre, de sintaxe amigável e contar com uma das maiores e mais ativas comunidades do mundo da tecnologia.

1.1 Motivação do trabalho

O desenvolvimento deste trabalho partiu do desejo de aprimoramento de competências chave, além da possibilidade de proporcionar um ganho real para a empresa analisada, o que consequentemente geraria uma visibilidade importante, visando crescimento profissional.

1.2 Justificativa

Em um mercado cada vez mais competitivo e as grandes exigências dos consumidores, a constante melhoria do nível de serviço se torna um conceito perseguido em todas as etapas do processo de entrega dos ativos. Assim, obter eficiência e eliminar desperdícios são fatores essenciais para manutenção da organização no mercado (BONELLI; FONSECA, 1998).

1.3 Objetivos

1.3.1 Objetivo Geral

Aprimorar a eficiência no processamento de devoluções em um centro de distribuição é fundamental, pois quanto mais ágil for a resolução do problema, maiores serão as possibilidades de reversão. Isso, por sua vez, refletirá em melhorias visíveis nos indicadores-chave analisados pela empresa.

1.3.2 Objetivos específicos

Dentre os objetivos específicos destacam-se:

- Mapear o processo de tratamento de devoluções;
- Entrevistar e entender juntamente com as partes interessadas o que tornaria o trabalho delas mais produtivo;
- Identificar os pontos de melhoria;
- Desenvolver e implementar uma solução RPA;
- Implementar a solução e treinar os funcionários para utilização do mesmo;
- Acompanhar os principais indicadores, pré e pós implementação.

2 REFERENCIAL TEÓRICO

2.1 Produtividade

A gestão da produtividade vem se fazendo cada vez mais necessária num ambiente de crescente abertura externa e globalização dos negócios. Sem produtividade e eficiência no processo produtivo, dificilmente empresas se tornam bem sucedidas, ou até mesmo sobrevivem no meio (MACEDO, 2012).

2.1.1 *Gestão da produtividade*

A gestão da produtividade incorpora basicamente 3 procedimentos, que são eles (MACEDO, 2012):

- A medição da produtividade;
- A identificação e a análise dos fatores determinantes dos gargalos de produtividade;
- A definição e aplicação de propostas de superação desses gargalos.

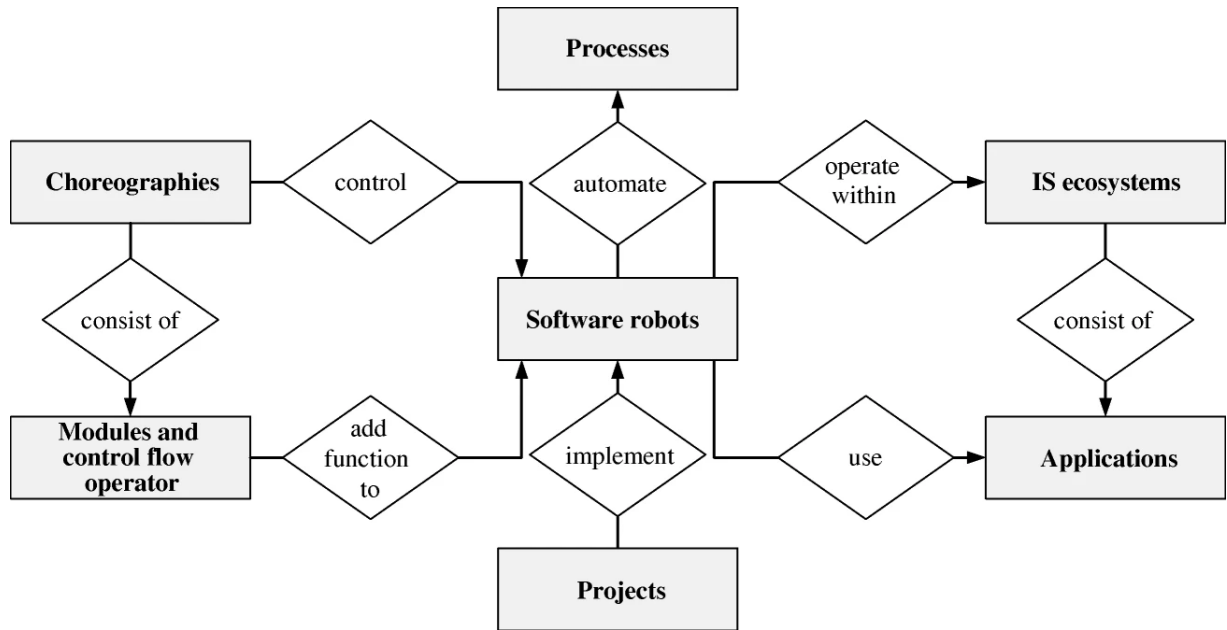
Nesse contexto, a medição da eficiência do processo produtivo torna-se determinada pela relação entre os bens e valores agregados (*output*) e os recursos utilizados pela empresa (*inputs*) no processo.

2.2 RPA - *Robotic Process Automation*

Para caracterizar o RPA de forma estruturada, suas principais características são apresentadas na figura 2.1. Esse entendimento é apresentado pela IEEE Corporate Advisory Group, que define RPA como:

O uso de um software pré-configurado que usa regras de negócios e coreografia de atividades predefinidas para completar a execução autônoma de uma combinação de processos, atividades, transações e tarefas em um ou mais sistemas de software não relacionados para entregar um resultado ou serviço com gerenciamento humano das exceções

Figura 2.1 – Natureza do processo de automação com utilização de robôs



Fonte: Hofmann et al. (2020)

RPA é uma abordagem para automatizar processos, dentro de um amplo conjunto de diferentes tecnologias para automação de processos em situações em que o trabalho humano, ou a construção e integração de sistemas de gerenciamento de processos de negócios (BPMS), são muito caros ou não justificados pelas necessidades do negócio. A RPA serve como um elemento de transição entre o trabalho humano e a ampla automação de processos de negócios (HOFMANN et al., 2020).

Assim, os chamados robôs acessam sistemas e realizam tarefas em sua maioria semelhantes aos humanos ou os imitando. A automação de processos por meio de RPA também pode se referir apenas à automação de atividades individuais ou mesmo tarefas (LACITY; WILLCOCKS, 2016).

Um exemplo de automação por meio de RPA, é a atividade cotidiana de uma secretária que, todos os dias as 9 da manhã atualiza uma planilha de *Excel*, e envia essas informações via *e-mail* para sua chefe. Essa tarefa poderia facilmente ser substituída por um robô automatizado, que a realizaria de maneira muito mais rápida e pontual, fazendo com que a secretaria possa empregar seu tempo em atividades de maior valor, e consequentemente proporcionar um melhor ambiente de trabalho.

2.3 Python

Python, dentre tantas outras opções de linguagem de programação, é extremamente interessante por se tratar de uma linguagem muito simples e clara, com uma sintaxe muito objetiva que vai direto ao ponto, o que a torna também uma boa opção para os que estão iniciando no mundo da programação (MENEZES, 2010). Graças ao trabalho da Python Foundation e de inúmeros colaboradores, é também uma linguagem de programação livre, o que significa que pode ser utilizado gratuitamente no Linux, Microsoft Windows ou Mac OS (MENEZES, 2010).

Pela forma que é estruturada, a linguagem é uma das mais utilizadas no que diz respeito a: análise de dados, *machine learning* e automação de processos, sejam eles processos *desktop* ou *Web* (MCKINNEY, 2019).

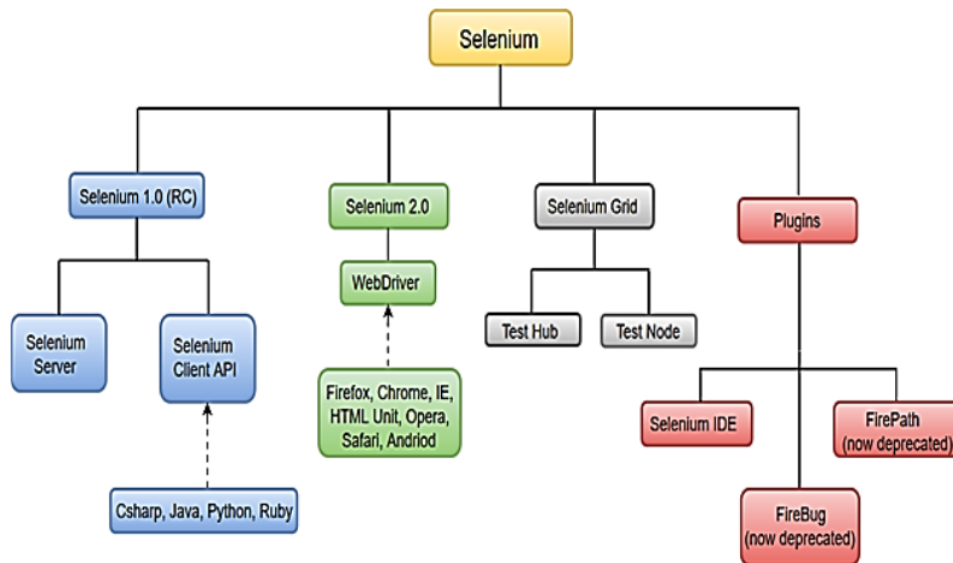
2.3.1 Selenium

Selenium é um *software* livre e suporta muitas plataformas como Linux, Windows e Mac. Selenium pode ser usado com navegadores como Google-chrome, Firefox e Internet-Explorer. Linguagens como: Java, C#, Python, Ruby podem ser usadas para o desenvolvimento de automações web ou criação de *softwares* de teste (SINGH et al., 2019). O Selenium é uma ferramenta de código aberto que se concentra na compatibilidade, podendo ser utilizado em uma variedade de navegadores, sendo eles:

- Google Chrome;
- Internet-Explorer 7, 8, 9, 10, e 11;
- Firefox;
- Safari;
- Opera;
- HtmlUnit;
- Phantomjs.

A figura 2.2 apresenta os principais componentes da ferramenta, vale salientar que o *selenium* já se encontra em sua versão 4.0.

Figura 2.2 – Componentes do Selenium



Fonte: Singh et al. (2019)

A funcionalidade utilizada será a descrita em verde: Selenium WebDriver. Esta consiste na abertura de uma instância de navegador (qualquer um dos citados acima) e é utilizada para simular interações humanas com o navegador, de maneira automatizada e extremamente mais rápida.

2.4 DDDM - Data driven decision making

Data driven decision making, ou tomada de decisões impulsionadas por dados, trata-se do uso de dados, fatos, e métricas para orientar decisões comerciais estratégicas, alinhadas com objetivos e métricas bem estabelecidas (BOUSDEKIS et al., 2021).

Quando se trata do processo de análise de dados, analisar dados para responder perguntas é parte essencial do processo, no entanto, existe um passo ainda mais importante, que é o agir (WESTBROOK, 1994).

3 METODOLOGIA

3.1 Contextualização e descrição do problema

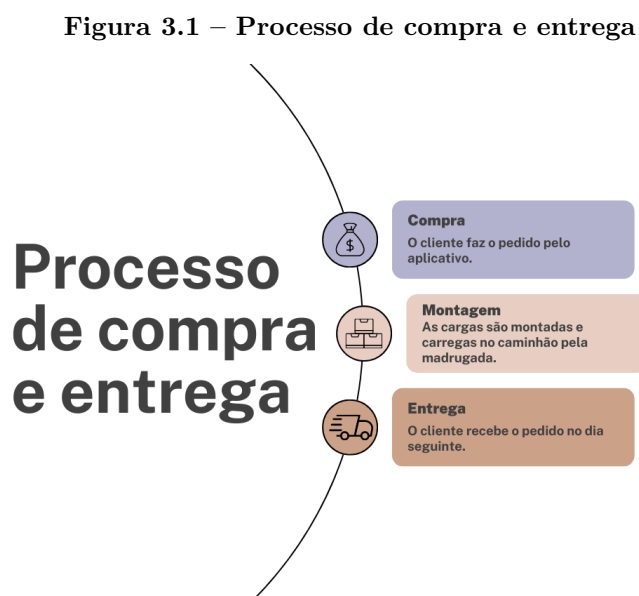
O presente trabalho tem como objetivo aumentar a produtividade e melhorar indicadores-chave do processo de entrega em um centro de distribuição de bebidas de grande porte, localizado na região metropolitana de Belo Horizonte. A empresa conta com cerca de 200 motoristas e ajudantes. Todos eles dispõem de um dispositivo móvel onde é instalado o aplicativo padrão para as entregas da empresa.

3.1.1 O aplicativo

No aplicativo os motoristas têm a visibilidade de todos os clientes a ser entregues no dia, assim como um GPS que os orienta a realizar entregas em uma sequência ótima, para melhor aproveitar o tempo em rota. Também contam com informações pertinentes de cada um dos clientes além de ser por intermédio do aplicativo a notificação de eventuais devoluções.

3.1.2 Sobre o processo de entrega

Em um cenário ideal os clientes fazem o pedido no aplicativo, a carga é montada a noite, e no outro dia recebem os produtos em seus estabelecimentos. A imagem 3.1 ilustra esta fase do processo:



Fonte: Próprio autor

3.1.3 Gestão da devolução

Como citado acima, em um cenário ideal, todos os pedidos realizados são entregues. Porém existem inúmeras barreiras que impedem que isso se concretize no panorama real.

Imagine o contexto geográfico em que esta empresa está inserida. Esta se encontra no Brasil, mais especificamente na cidade de Belo Horizonte, cidade caracterizada por morros e vales, ruas estreitas e acrescentada da complexidade de se entregar em regiões mais afastadas, como as comunidades nos morros. Neste cenário é comum não se encontrar vagas para estacionamento de veículos de grande porte, que não seja possível entrar em determinadas ruas porque ela é muito ingrime ou estreita.

Muitos dos problemas encontrados em rota não são resolvíveis, no entanto, se o problema fosse: o motorista chegou ao estabelecimento mas o mesmo estava fechado; nesse caso a equipe de entrega pode notificar esse motivo de devolução no aplicativo, e o monitoramento - equipe encarregada de realizar as tratativas para as as devoluções - tentará resolver. O fluxo se dá como na imagem 3.2:

Figura 3.2 – Comunicação de devoluções



Fonte: Próprio autor

A parte do processo que demanda a maior mão de obra, e que conseqüentemente traz maiores oportunidades de melhoria, se dá a partir do apontamento no aplicativo, onde o monitoramento inicia sua rotina para resolução das devoluções. A figura 3.3 descreve o processo:

Figura 3.3 – Processo de tratativa de devoluções



Fonte: Próprio autor

Nas etapas de captação, tratativa e visibilidade existem diversos gargalos que fazem com o processo seja pouco eficiente. Se tratam de tarefas manuais que não agregam valor ao processo e apenas o tornam mais demorado, dentre essas tarefas cabe destacar:

- A plataforma não notifica o monitoramento quando há novas devoluções;
- A plataforma é instável, e em alguns casos pode demorar até 5 minutos entre o horário que a devolução foi apontada e o tempo em que ela é indicada na plataforma;
- Não existe um local padronizado onde o monitoramento tenha acesso a todos os contatos necessários: número atualizado do cliente, número do motorista, e número do vendedor. Com isso perde-se tempo navegando entre sistemas;
- É necessário que o monitoramento copie e cole informações da plataforma em grupos de Whatsapp que dão visibilidade das devoluções à equipe de vendas.

3.2 Tempo de captação

Para medir a eficácia da intervenção, serão utilizados dois parâmetros importantíssimos que são utilizados para medir a produtividade produtividade da ação do monitoramento.

O tempo de captação é medido a partir da diferença entre o horário em que se iniciou a tratativa, e o horário que a devolução foi sinalizada no aplicativo, como se observa na equação 1 :

$$TC = HT - HA \quad (1)$$

Onde:

- TC : Tempo de captação
- HT : Hora de tratamento
- HA : Hora de apontamento

Diminuir o tempo de captação essencialmente também significa aumentar as chances de reversão (outro indicador que será explicado posteriormente). Uma demora muito grande para descobrir o problema pode significar perda da janela de oportunidade para resolução da devolução.

3.3 Reversão

A reversão é o indicador mais importante se fala do trabalho que o monitoramento exerce. Além de servir como incentivo em forma de remuneração variável, esse número expressa a eficácia com que as devoluções estão sendo tratadas.

Esse indicador diz respeito a quantidade de devoluções efetivamente revertidas após atuação do monitoramento, se comunicando com as partes interessadas e realizando os alinhamentos para o recebimento da mercadoria. E é calculado como se vê na equação 2:

$$RV = \frac{DV}{DT} \quad (2)$$

Onde:

- RV : Taxa de reversão;
- DV : Devoluções revertidas;

- *DT* : Devoluções totais.

Esse número está estritamente relacionado ao tempo de captação, uma vez que a celeridade nas tratativas de devolução aumentam sua taxa de sucesso.

3.3.1 Bate-papo com o monitoramento

Para sair com um plano de ação bem estruturado, se faz necessária uma conversa com a equipe de monitoramento, pois são estes que estão na linha de frente da operação, melhorar o processo de tratativa de devolução significa não só um ganho de produtividade, como também uma melhoria nas suas condições de trabalho.

Nesta conversa lhes foi apresentado um escopo do que era a ideia inicial do projeto, e como era imaginado que o mesmo poderia ajudar em suas tarefas. A partir daí foram passados diversos *insights* de melhoria, alguns destes muito factíveis, que acabaram por fazer parte do projeto final. Dentre estes cabe citar:

- Conter na mensagem, todos os números do cliente disponíveis no sistema;
- Elencar os números das notas fiscais sendo devolvidas, para que seja acelerado um processo particular da tratativa da devolução;
- Enviar as notificações em grupo geral no horário de almoço.

3.4 Desenvolvimento da solução RPA

Considerando todos os gargalos citados acima, faz-se necessário a implementação de uma ferramenta que possa realizar boa parte das tarefas manuais. Para isso, deve-se fazer um apanhado de todas as informações estatísticas necessárias para realização do projeto.

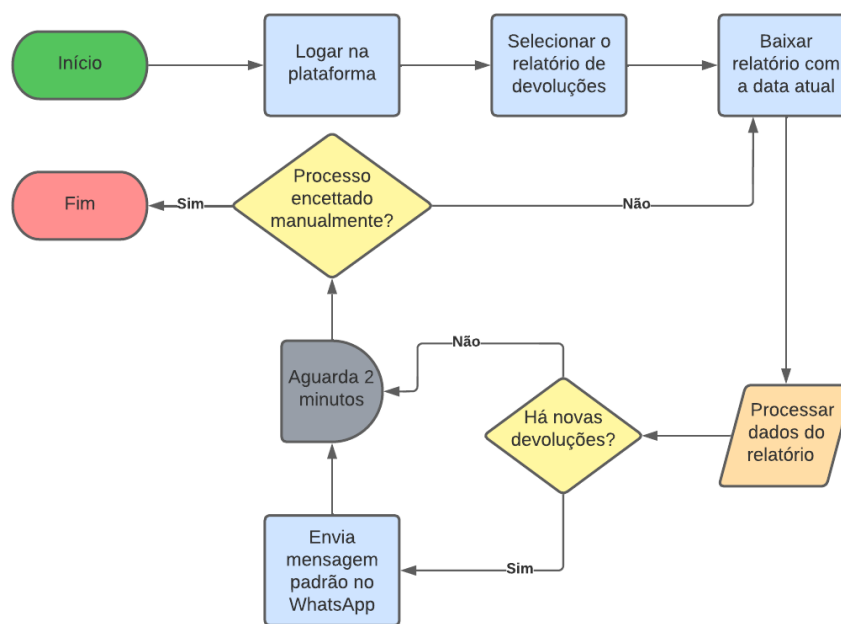
Foram então organizadas informações como: todos os números disponíveis para cada cliente no sistema, os números de todos os vendedores, os contatos dos motoristas e de seus supervisores diretos, assim como a equipe que cada um desses motoristas pertence. Essas informações não dependem de uma atualização periódica e serão consultadas sempre que necessário.

Por se tratar de uma plataforma que gera informações para todas as ações realizadas, que conseqüentemente são armazenadas em relatórios, é possível iniciar uma instância de navegador utilizando *selenium* para acessar a página Web da plataforma e

realizar o *download* periódico desses relatórios. Dessa forma, ao analisar as informações contidas no arquivo csv, é possível determinar: quando a uma nova devolução, o motorista que está devolvendo, a placa do veículo, o cliente e o motivo da devolução.

Com essas informações em mãos, é possível cruzá-las com as informações previamente armazenadas e elaborar mensagens de texto personalizadas, que serão posteriormente enviadas em grupos específicos de *WhatsApp*. O fluxograma a seguir ilustra o processo de forma mais organizada:

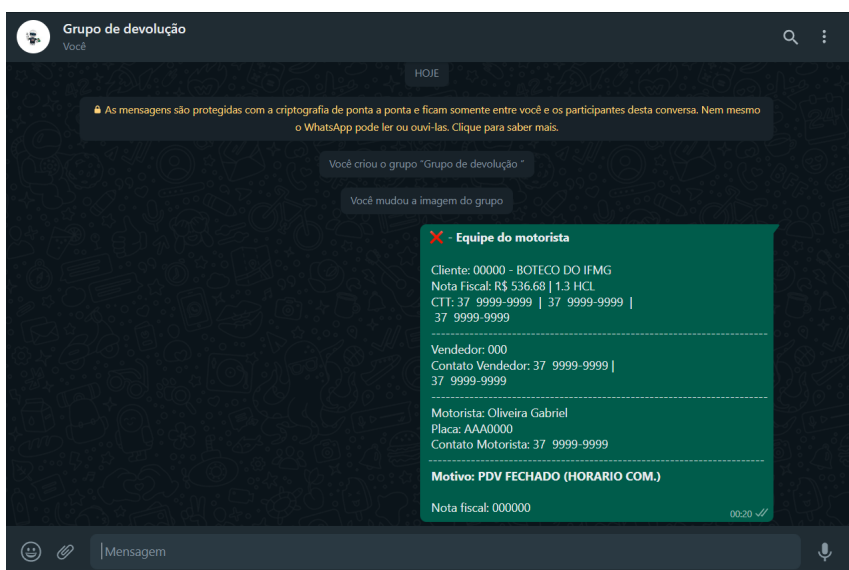
Figura 3.4 – Processo de envio de mensagens automatizadas



Fonte: Próprio autor

A mensagem padrão enviada contém todas as informações necessárias que irão facilitar a resolução das devoluções, com a vantagem de que agora o monitoramento é notificado toda vez que há uma nova devolução, e todas as informações já estão em mãos, sem a necessidade de pesquisar nenhuma delas em outro sistema ou plataforma. A imagem 3.5 mostra como a mensagem é estruturada:

Figura 3.5 – Mensagem padrão



Fonte: Próprio autor

Os caracteres em azul são reconhecidos pelo WhatsApp como números de telefone, consequentemente é possível ligar para o conta ou enviar mensagens a partir dessa mesma página, apenas clicando sobre o número, sem a necessidade de adicionar o número na lista de contatos, ou digitá-lo no discador, economizando tempo precioso que poderá ser utilizado em outras atividades.

Cada funcionário do monitoramento é responsável por acompanhar equipes específicas, então foram criados grupos para cada uma das equipes, neles contidos o monitoramento responsável por aquela equipe, assim como o supervisor da equipe.

3.5 Subprocessos importantes

Como a automação deverá ser utilizada por pessoas que não necessariamente entendem de programação, seu uso deve ser extremamente facilitado, e para isso foi criados outro subprocesso dentro do processo de envio das mensagens automáticas.

3.5.1 Armazenar devoluções já tratadas

Em caso de erro em qualquer uma das etapas do processo anterior, o usuário deveria reiniciar o *software*, e nesse caso, todas as informações salvas anteriormente seriam perdidas, e o *software* enviaria mais uma vez, todas as devoluções reportadas naquele dia.

Para evitar envio e reenvio de mensagens desnecessárias ao longo do dia, deveria haver alguma forma de obter uma lista com todas as devoluções já reportadas. Então foi

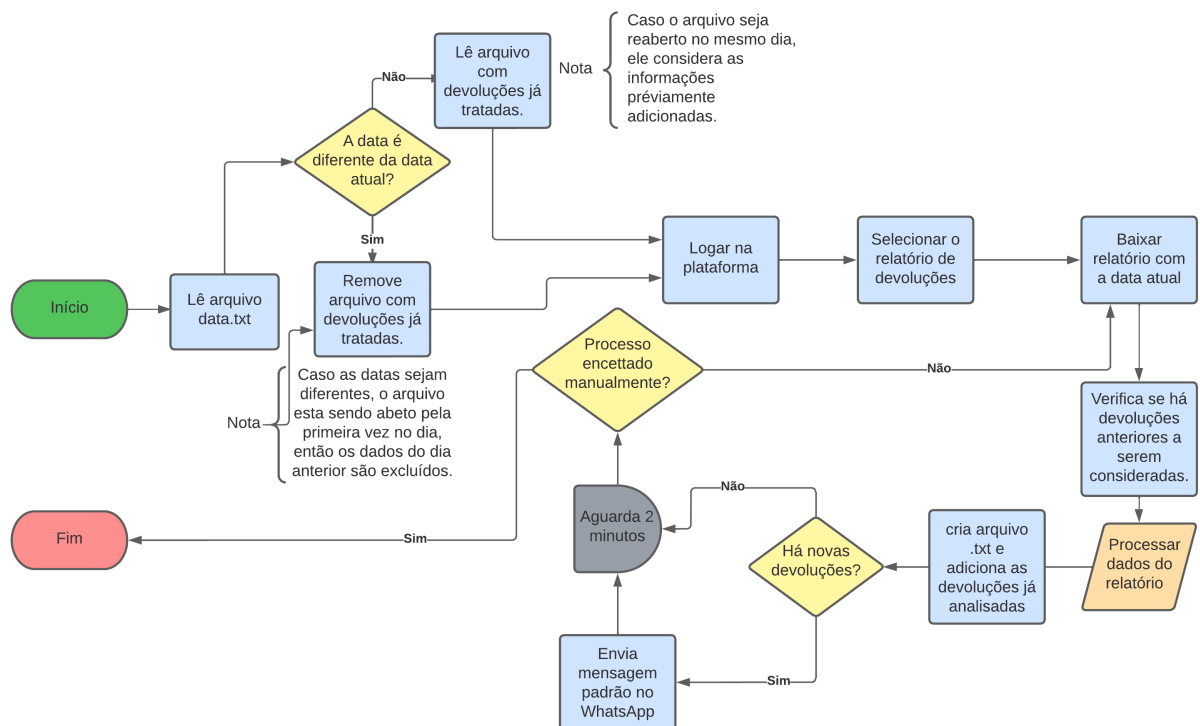
adicionada uma funcionalidade que, ao final do processo de "Processamento de dados do relatório", fosse criado um arquivo de texto contendo todas as devoluções já tratadas.

A partir desse ponto, sempre que o *software* é iniciado ele deve ler este arquivo de texto e verificar quais linhas devem ser ignoradas ao baixar o relatório novamente.

Isso por si só resolve um dos problemas encontrados. No entanto, se em um outro dia o *software* fosse iniciado, ele carregaria as informações relativas ao dia anterior, o que faria com que o processo ignorasse devoluções que não deveriam ser ignoradas, e nesse caso, os dados referentes ao dia anterior deveriam ser apagados manualmente.

Como o intuito é a utilização do *software* seja a mais simples possível, e que o fator "erro humano" seja desconsiderado do processo, faz-se necessário também automatizar o processo de checagem da data em que o programa está funcionando. Dessa forma, a lógica utilizada para resolução do problema é ilustrada no fluxograma abaixo:

Figura 3.6 – Fluxograma atualizado



Fonte: Próprio autor

Com o fluxo rodando de forma que seja necessária uma mínima interferência humana, foi criado um executável "dev.exe", pois dessa forma pode-se agendar a execução do *script* diariamente, e em caso de falha o operador deve apenas executar o programa pressionando o botão "Enter".

4 RESULTADOS

Após término do projeto as equipes foram treinadas sobre a utilização da ferramenta e foram coletados dados de 2 meses após o *software* começar a ser utilizado. Esses dados foram confrontados com as informações obtidas antes de sua utilização, quando se usava o fluxo de tratativas antigo.

4.0.1 Comparação dos tempos de captação

Para esta análise foram captados dados de relatórios gerados a partir do uso da plataforma, considerando o período vigente entre janeiro e agosto de 2022.

A tabela apresentada na imagem 4.1 exibe o tempo médio de captação de cada um dos operadores do monitoramento, seus nomes foram censurados por serem informações sensíveis. As colunas representam cada um dos operadores, e nas linhas são apresentados o tempo médio de captação de cada um considerando o mês do ano, indo de 1 (janeiro), até 8 (agosto).

Figura 4.1 – TC pré e pós ferramenta

		Média Tempo de Captação							
Nome/Mês	Operador 1	Operador 2	Operador 3	Operador 4	Operador 5	Operador 5	Operador 6	GERAL	
1	-	00:16:20	00:17:14	00:11:06	-	-	-	00:18:49	
2	-	00:24:02	00:23:34	00:17:24	-	-	00:13:00	00:21:32	
3	-	00:26:27	00:34:34	00:17:27	00:27:27	-	00:11:48	00:25:27	
4	-	00:39:23	00:39:45	00:10:07	00:37:33	-	00:17:10	00:35:36	
5	-	00:23:56	00:30:53	00:20:39	00:34:26	-	00:16:56	00:27:08	
25-jun	6	-	00:22:47	00:35:45	00:16:28	00:22:56	00:25:15	00:30:44	00:23:22
	7	00:14:25	00:15:10	00:19:26	00:09:58	00:13:34	00:10:13	00:31:19	00:14:28
	8	00:27:14	00:19:49	00:24:56	00:13:21	00:12:38	00:13:41	00:06:48	00:18:06

Fonte: Próprio autor

A ferramenta passou a ser utilizada a partir do dia 25 de junho. A partir desse ponto é possível notar uma diminuição drástica no tempo necessário para tratamento das devoluções. Quando observados individualmente, verifica-se que a produtividade de cada um dos operadores aumentou de forma expressiva, em alguns casos, devoluções estavam sendo tratadas até 50% mais rápidas.

4.0.2 Comparação reversão

Fazendo uso da mesma base de dados utilizada para o cálculo do tempo de captação, foi possível determinar a quantidade média de devoluções revertidas em cada

dia. Essas foram então comparadas mês a mês, para que fosse possível verificar melhora no indicador após uso da ferramenta, em caso de alguma melhora. A tabela na figura 4.2 demonstra a evolução desses números ao longo do ano:

Figura 4.2 – Taxa de reversão x meses do ano

Média diária reversão de devolução		
Mês	Qtd	%
1	10	33,01%
2	26	26,04%
3	23	19,38%
4	23	20,97%
5	25	28,85%
6	22	24,28%
7	33	40,33%
8	34	35,66%

Fonte: Próprio autor

Ao analisar as informações contidas na tabela, verifica-se um aumento substancial na quantidade de reversões realizadas diariamente. Nota-se também que a porcentagem de reversões relativas a quantidade de devoluções no dia, passou a representar uma parcela bem maior do todo. Em números, esse aumento representou em média, um ganho de aproximadamente 45% no número de devoluções revertidas diariamente.

4.0.3 Escala do monitoramento

O monitoramento é atualmente composto por 7 indivíduos, e cada um deles inicia seu turno em horários diferentes para que seja possível ter pessoas acompanhando devoluções até as 22 horas. No entanto a forma com que a escola foi imaginada se deu pela intuição de seus antigos supervisores.

Por acreditar que havia uma demora maior de atendimento entre as 11 e as 14 horas devido o horário de almoço, foi estratificado o tempo de captação ao longo do dia, para que fosse possível entender em que momento seria necessário agir. A figura 4.3 descreve a forma com esse movimento se dá com o passar das horas:

Figura 4.3 – Tempo de captação ao longo do dia

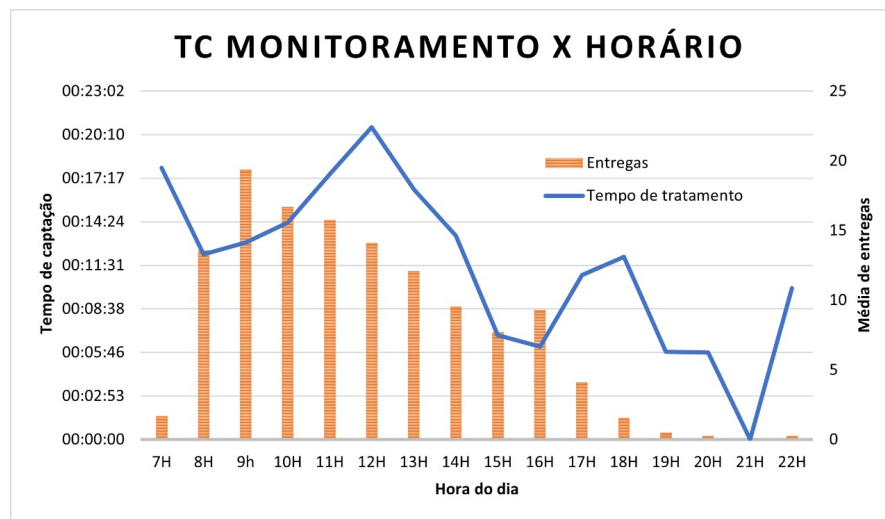
Hora do dia/Operador	7H	8H	9h	10H	11H	12H	13H	14H	15H	16H	17H	18H	19H	20H	21H	22H
Operador 1	-	-	-	-	-	-	-	-	-	00:04:20	-	00:03:00	-	00:05:00	00:17:00	00:07:40
Operador 2	00:14:57	00:14:13	00:17:53	00:14:42	00:23:54	00:26:10	00:15:25	00:12:31	00:07:57	00:06:49	00:13:23	00:13:22	00:05:17	00:07:45	00:01:00	00:17:00
Operador 3	00:13:24	00:13:32	00:15:30	00:18:35	00:24:25	00:49:52	00:20:53	00:14:17	-	-	-	-	-	-	-	-
Operador 4	-	00:06:25	00:07:51	00:11:01	00:11:54	00:13:26	00:15:40	00:08:34	00:06:20	00:06:00	-	-	-	-	-	-
Operador 5	-	00:33:46	00:13:10	00:06:09	00:08:26	00:22:35	00:24:17	00:20:48	00:06:20	00:03:00	-	-	-	-	-	-
Operador 6	-	00:17:54	00:21:22	00:10:51	00:07:09	00:06:58	00:12:40	00:13:14	00:06:38	00:05:20	00:06:22	00:03:00	-	-	-	-
Operador 7	00:41:00	00:28:20	00:47:36	00:23:43	00:32:00	00:32:26	00:18:00	-	-	-	-	-	-	-	-	-
Geral	00:16:38	00:15:09	00:15:48	00:14:23	00:16:46	00:19:47	00:16:22	00:12:55	00:07:24	00:06:34	00:11:41	00:11:27	00:05:53	00:05:34	00:09:00	00:10:00

Fonte: Próprio autor

Por ser feito um rodízio entre os integrantes do monitoramento, dentre aqueles que vão pela manhã, todos apresentam um TC na maioria dos horários da manhã. Como pode ser observado, a parte da manhã é de fato a mais improdutiva, sendo constatado um pico as 12 horas, quando muitos estão em horário de almoço.

Para entender a causa raiz desse problema, faz-se necessária uma visualização que consiga mostrar também a necessidade de mão de obra ao longo do dia. O gráfico na figura 4.4 esboça o tempo necessário para tratar devoluções ao longo do dia, confrontando-o com a quantidade de entregas que requerem atenção naquele dado momento:

Figura 4.4 – Tempo de captação x numero de devoluções ao longo do dia



Fonte: Próprio autor

Como pode ser verificado, boa parte das devoluções ocorrem até as 12 horas, o que significa esse período exige mais atenção daqueles cuja tarefa é resolver esse tipo de problema.

Em posse dessas informações, foi apresentado um plano de escala mais otimizado aos supervisores do monitoramento. Após a conversa foi possível elaborar uma escala que atenderia melhor a necessidade de mão de obra humana naquele período e ainda respeitasse

particularidades de cada um dos colaboradores, como distância e tempo até chegar ao trabalho, assim como a periculosidade de determinadas regiões a partir das 19 horas.

5 CONCLUSÃO

Ao final do processo de elaboração e implementação da ferramenta de automatização, verificou-se melhoria substancial na qualidade das condições de trabalho da equipe de monitoramento. Conseqüentemente foi possível observar uma melhora significativa nas métricas que determinam a produtividade e a eficácia do trabalho dessa equipe.

O tempo de captação, que reflete o tempo necessário para se iniciar as tratativas das devoluções, apresentou uma melhora de 40%. Em teoria, diminuir esse tempo também significaria aumentar a quantidade de devoluções revertidas, uma vez que tratar os problemas mais rápido pode facilitar as suas resoluções, em um cenário logístico tão complexo. Após análise, a expectativa foi concretizada. A taxa de reversão apresentou melhora na casa dos 45%, o que representou um aumento de cerca de 8 a 9 devoluções revertidas a cada dia.

Ao se analisar a fundo dados provenientes da rota, foi possível enxergar diversos *gaps*, que caso resolvidos poderiam significar também uma grande melhora nos indicadores. A principal dessas oportunidades diz respeito a escala de horários dos integrantes do monitoramento, que ao analisar a necessidade de mão de obra humana ao longo do dia, foi identificado que existia uma escala melhor, que se encaixaria melhor para a demanda da manhã e ainda respeitaria as necessidades individuais dos colaboradores.

Um ponto importante é que a remuneração variável do monitoramento tem como parâmetros os indicadores analisados, o que significa que essa melhora representará também mais dinheiro e melhor qualidade de vida para esses colaboradores. No fim, um ponto importante que muitas vezes passa despercebido, é o papel da academia e da engenharia na vida das pessoas, e da sociedade como um todo. A inovação tecnológica e os ganhos de produtividade vivenciados nos últimos séculos, fizeram com que a sociedade se encontrasse em um patamar inimaginável. Com isso, desenvolver soluções que tornam processos: mais rápidas, mais eficientes, maiores etc; quase em sua totalidade significam também melhorar a vida das pessoas, ao invés de apenas "maiores lucros" para grandes empresas.

REFERÊNCIAS

- ABREU, C. E. M.; GONZAGA, D. R. B.; SANTOS, F. J. dos; OLIVEIRA, J. F. de; OLIVEIRA, K. D. de M.; FIGUEIREDO, L. M.; NASCIMENTO, M. P.; OLIVEIRA, P. G. de; YOSHINAGA, S. T. de S.; OLIVEIRA, T. T. de et al. Indústria 4.0: Como as empresas estão utilizando a simulação para se preparar para o futuro. **Revista de Ciências Exatas e Tecnologia**, v. 12, n. 12, p. 49–53, 2017.
- BONELLI, R.; FONSECA, R. Ganhos de produtividade e de eficiência: novos resultados para a economia brasileira. Instituto de Pesquisa Econômica Aplicada (Ipea), 1998.
- BOUSDEKIS, A.; LEPENIOTI, K.; APOSTOLOU, D.; MENTZAS, G. A review of data-driven decision-making methods for industry 4.0 maintenance applications. **Electronics**, MDPI, v. 10, n. 7, p. 828, 2021.
- HOFMANN, P.; SAMP, C.; URBACH, N. Robotic process automation. **Electronic Markets**, Springer, v. 30, n. 1, p. 99–106, 2020.
- IVANČIĆ, L.; VUGEC, D. S.; VUKŠIĆ, V. B. Robotic process automation: systematic literature review. In: SPRINGER. **International Conference on Business Process Management**. [S.l.], 2019. p. 280–295.
- LACITY, M.; WILLCOCKS, L. Paper 16/01 robotic process automation: The next transformation lever for shared services. **Retrieved from The Outsourcing Unit, LSE: <http://www.umsl.edu/lacitym>**, 2016.
- MACEDO, M. de M. Gestão da produtividade nas empresas. **Revista Organização Sistêmica**, v. 1, n. 1, p. 110–119, 2012.
- MCKINNEY, W. **Python para análise de dados: Tratamento de dados com Pandas, NumPy e IPython**. [S.l.]: Novatec Editora, 2019.
- MENEZES, N. N. C. Introdução a programação com python. **São Paulo: Novatec**, 2010.
- SINGH, D.; BABU, M. R.; PRASHAR, D.; RAKHRA, M. Software testing with selenium web driver using python. **Think India Journal**, v. 22, n. 30, p. 300–305, 2019.
- TUPY, O.; YAMAGUCHI, L. C. T. Eficiência e produtividade: conceitos e medição. **Agricultura de São Paulo, Sao Paulo**, v. 45, n. 2, p. 39-51, 1998., 1998.
- WESTBROOK, L. Qualitative research methods: A review of major stages, data analysis techniques, and quality controls. **Library & information science research**, Elsevier, v. 16, n. 3, p. 241–254, 1994.

APÊNDICE A – CÓDIGO FONTE

Abaixo segue o código fonte do projeto. Em muitas etapas poderia ter sido utilizado de melhores práticas para desenvolvimento do *software*, no entanto foi utilizada uma abordagem prática e funcional. Informações sensíveis foram censuradas.

```
1 import glob
2 import warnings
3 warnings.simplefilter(action='ignore', category=FutureWarning)
4 import pandas as pd
5 import os
6 from selenium import webdriver
7 from selenium.webdriver.chrome.options import Options
8 from webdriver_manager.chrome import ChromeDriverManager
9 from selenium.webdriver.chrome.service import Service
10 from selenium.webdriver.common.keys import Keys
11 import time
12 from datetime import datetime
13 import zipfile
14 from pathlib import Path
15
16 # Leitura da data do arquivo 'reset.txt'
17 with open('reset.txt') as f:
18     data = f.readlines()
19
20 data = data[0]
21 data = datetime.strptime(data, r'%d/%m/%Y')
22
23 # Obtm a data e hora atual
24 hoje_datetime = datetime.today()
25
26 # Se o dia da ltima execucao for diferente do dia atual, remove o arquivo 'pdvs_tratados.
    csv'
27 if data.day < hoje_datetime.day:
28     try:
29         os.remove('pdvs_tratados.csv')
30     except:
31         pass
32
33 # Atualiza o arquivo 'reset.txt' com a data atual
34 hoje = datetime.today().strftime(r'%d/%m/%Y')
35 with open('reset.txt', 'w') as f:
36     f.write(hoje)
37
38 # Caminho para a pasta de download e diretorio atual
39 path_to_download_folder = str(os.path.join(Path.home(), "Downloads"))
```

```

40 cwd = os.getcwd()
41
42 # Definio de filtros
43 filtro_apontamento = ['Veiculo', 'Motorista', 'Valor Devolucao (R$)', 'Tipo de
    Apontamento', 'Cliente',
44                        'Motivo Motorista', 'Vendedor', 'Mapa', 'Vol. HCL', 'Cd. Cliente'
    ]
45 filtro_aderencia = ['Mapa', 'Cod. Cliente', 'Chegada ao PDV', 'Repasse', 'PDV Lacrado',
    'Acessou raio PDV?',
46                    'Ocorrncia', 'Tempo no PDV']
47 filtro_consolidado = ['Veiculo', 'Motorista', 'Supervisor Rota', 'Valor Devolucao (R$)', '
    Cliente', 'Motivo Motorista',
48                    'Vendedor', 'Sala', 'Mapa_x', 'Vol. HCL', 'Chegada ao PDV', '
    Repasse', 'PDV Lacrado',
49                    'Acessou raio PDV?', 'Tempo no PDV']
50
51 # Dicionrios para mapear valores
52 supervisores = {0: 'Inativo', 1: 'Noturna', 2: 'Van CDL', 3: 'Raa', 4: 'Bravos', 5: '
    Elite', 6: 'Fora', 7: 'Freteiro',
53                 8: 'Alfa', 9: 'UDC', 10: 'N&LOG'}
54 grupos = {0: 'DEV - Freteiros', 1: 'DEV - Noturna', 2: 'DEV - Vans', 3: 'DEV - Raa', 4:
    'DEV - Bravos',
55           5: 'DEV - Elite', 6: 'DEV - Fora', 7: 'DEV - Freteiros', 8: 'DEV - Alfa ', 9:
    'DEV - Vans',
56           10: 'DEV - Freteiros'}
57
58 # Leitura do arquivo de motoristas
59 motoristas = pd.read_csv(cwd + r'\depara\motoristas.csv', sep=';', encoding='latin1')
60 motoristas['Nome'] = motoristas['Nome'].str.rstrip()
61 motoristas['Nome'] = motoristas['Nome'].str.upper()
62
63 # Funo para baixar o relatrio de apontamento
64 def baixar_apontamento():
65     time.sleep(0.5)
66     try:
67         navegador.get('https://ambev.tracking2nd.com.br/Relatorios/
    ApontamentoDevolucao')
68     except:
69         baixar_apontamento()
70
71     # ... (restante da funo)
72
73 # Funo para baixar o relatrio de aderencia
74 def baixar_aderencia():
75     time.sleep(0.5)
76     navegador.get('https://ambev.tracking2nd.com.br/Relatorios/AderenciaCustom')
77

```

```

78     # ... (restante da funo)
79
80 # Funo para abrir o WhatsApp Web
81 def abrir_zap():
82     print('''-----
83     Por favor, leia o QR Code e conecte-se ao WhatsApp...
84     ''')
85     navegador.get('https://web.whatsapp.com/')
86
87     # ... (restante da funo)
88
89 # Funo para remover arquivos da pasta de download
90 def remover_arquivos_download():
91     list_of_files = glob.glob(f'{path_to_download_folder}/*.csv')
92     for file in list_of_files:
93         os.remove(file)
94
95 # Funo para remover arquivos
96 def remover_arquivos():
97     list_of_csv_files = glob.glob(cwd + r'\\Apontamento\\*.csv')
98
99     # ... (restante da funo)
100
101 # instanciando navegador
102
103 chrome_options = Options()
104 servico = Service(ChromeDriverManager().install())
105 navegador = webdriver.Chrome(service=servico, options=chrome_options)
106
107 # Entrar No whats
108 abrir_zap()
109 pWindow = navegador.current_window_handle
110
111 # LOGIN
112 navegador.execute_script("window.open('https://ambev.tracking2nd.com.br/');")
113 handles = navegador.window_handles
114 newHandle = handles[1]
115 navegador.switch_to.window(newHandle)
116
117 # credenciais
118 usuario = navegador.find_element('id', 'Usuario')
119 senha = navegador.find_element('id', 'Senha')
120 usuario.send_keys('*****')
121 senha.send_keys('***')
122 navegador.find_element('xpath', '//*[@id="form0"]/div[2]/input[3]').click()
123
124 # lista com pdvs j enviados

```

```

125 try:
126     pdvs_tratados = pd.read_csv('pdvs_tratados.csv', encoding='latin1', sep=';')
127     pdvs_tratados = pd.read_csv('pdvs_tratados.csv', encoding='latin1', sep=';')
128     pdvs_tratados = pdvs_tratados.iloc[:, 0]
129     pdvs_tratados = list(pdvs_tratados)
130 except:
131     pdvs_tratados = []
132
133 with open('reset.txt', 'w') as f:
134     f.write(hoje)
135
136 while True:
137     # Baixar relatrios
138     baixar_apontamento()
139     baixar_aderencia()
140
141     try:
142         # Mover arquivos da pasta de download
143         list_of_zip_files_download = glob.glob(f'{path_to_download_folder}\\*.zip')
144         latest_zip_file = max(list_of_zip_files_download, key=os.path.getctime)
145         os.replace(latest_zip_file, cwd + r'\Aderencia\AderenciaAmBev.zip')
146
147     except ValueError:
148         print('-----')
149         print('Houve um erro ao baixar o *Aderencia Ambev*...')
150         print('Excluindo arquivos...')
151         print('*REINICIANDO PROCESSO*')
152         remover_arquivos_download()
153         continue
154
155     # Mover CSVs
156     list_of_csv_files_download = glob.glob(f'{path_to_download_folder}\\*.csv')
157     latest_csv_file = max(list_of_csv_files_download, key=os.path.getctime)
158     os.replace(latest_csv_file, cwd + r'\Apontamento\ApontamentoDevolucao.csv')
159
160     # Descompactar arquivo ZIP
161     with zipfile.ZipFile(f'{cwd}\Aderencia\AderenciaAmBev.zip', 'r') as zip_ref:
162         zip_ref.extractall(f'{cwd}\Aderencia')
163
164     # Tratar dados
165     apontamento = pd.read_csv(cwd + r'\Apontamento\ApontamentoDevolucao.csv', sep=';',
166                               encoding='latin1', usecols=filtro_apontamento)
167     aderencia = pd.read_csv(cwd + r'\Aderencia\AderenciaAmBev.csv', sep=';', encoding='latin1', usecols=filtro_aderencia)
168
169     # Filtrar dados relevantes
170     apontamento = apontamento[(apontamento['Tipo de Apontamento'] == 'Devolucao')]

```

```

170
171     try:
172         # Tratar formato da coluna 'Valor Devoluo (R$)'
173         apontamento['Valor Devoluo (R$)'] = apontamento['Valor Devoluo (R$)'].str.
replace('.', '')
174         apontamento['Valor Devoluo (R$)'] = apontamento['Valor Devoluo (R$)'].str.
replace(',', '.')
175         apontamento['Valor Devoluo (R$)'] = apontamento['Valor Devoluo (R$)'].astype(
float)
176
177     except:
178         print('-----')
179         Ainda no h devolues a serem enviadas! E que continue assim, amm!''')
180         time.sleep(90)
181         print('*REINICIANDO PROCESSO*')
182
183     # Processamento adicional dos dados
184     apontamento_no_duplicates = apontamento.sort_values('Cd. Cliente').drop_duplicates
(subset='Cd. Cliente').reset_index(drop=True)
185     coluna_soma = apontamento[['Cd. Cliente', 'Valor Devoluo (R$)']].sort_values('Cd.
Cliente').groupby('Cd. Cliente', as_index=False).sum()['Valor Devoluo (R$)']
186     apontamento_no_duplicates['Soma'] = coluna_soma
187     apontamento_no_duplicates['Vendedor'] = apontamento_no_duplicates['Vendedor'].str
[:3]
188     apontamento_no_duplicates['Sala'] = apontamento_no_duplicates['Vendedor'].str[0].
astype(int)
189     apontamento_no_duplicates.sort_values('Sala', inplace=True)
190     apontamento_no_duplicates.reset_index(drop=True, inplace=True)
191     apontamento_no_duplicates['Chave'] = apontamento_no_duplicates['Mapa'].astype(str)
+ apontamento_no_duplicates['Cd. Cliente'].astype(str)
192     apontamento_no_duplicates['Motorista'] = apontamento_no_duplicates['Motorista'].
str.upper()
193
194     # Processamento adicional dos dados de aderencia
195     aderencia['Cod. Cliente'] = aderencia['Cod. Cliente'].str.replace("-", "0")
196     aderencia['Cod. Cliente'] = aderencia['Cod. Cliente'].str[-5:]
197     aderencia.dropna(subset='Cod. Cliente', inplace=True)
198     aderencia['Cod. Cliente'] = aderencia['Cod. Cliente'].astype(int)
199     aderencia = aderencia[aderencia['Ocorrncia'] == 'Devoluo']
200     aderencia['Chave'] = aderencia['Mapa'].astype(str) + aderencia['Cod. Cliente'].
astype(str)
201     aderencia.drop_duplicates(subset='Cod. Cliente', inplace=True)
202
203     # Merge dos dados
204     consolidado = pd.merge(apontamento_no_duplicates, aderencia, on='Chave', how='left'
)
205     consolidado = pd.merge(consolidado, motoristas, left_on='Motorista', right_on='

```

```

Nome', how='left')
206 consolidado['Sala Rota'] = consolidado['Supervisor Rota']
207 consolidado['Supervisor Rota'] = consolidado['Supervisor Rota'].map(supervisores)
208 consolidado['Supervisor Rota'] = consolidado['Supervisor Rota'].fillna('')
209 consolidado['Sala Rota'] = consolidado['Sala Rota'].fillna('7')
210 consolidado = consolidado[~consolidado['Cd. Cliente'].isin(pdvs_tratados)]
211
212 # Atualizar lista de PDVs tratados
213 pdvs_tratados = list(pdvs_tratados) + list(consolidado['Cd. Cliente'])
214 csv_pdvs = pd.Series(pdvs_tratados)
215 csv_pdvs.to_csv('pdvs_tratados.csv', index=False, encoding='latin1', sep=';')
216
217 # Mensagens de log com base no resultado do processamento
218 if len(consolidado['Cd. Cliente']) == 0:
219     print(''-----
220           Base tratada.
221           No houve devolues no ltimo minuto! Graas a Deus pai, S!
222           *REINICIANDO PROCESSO*''')
223
224 elif len(consolidado['Cd. Cliente']) == 1:
225     print(f''-----
226           Base tratada.
227           Houve {len(consolidado['Cd. Cliente'])} devoluo no ltimo minuto.
228           Enviando resultado...
229           *REINICIANDO PROCESSO*''')
230
231 else:
232     print(f''-----
233           Houve {len(consolidado['Cd. Cliente'])} devolues no ltimo minuto.
234           Enviando resultados...''')
235
236 # Configurao para alternar entre janelas do navegador
237 navegador.switch_to.default_content()
238 navegador.switch_to.window(pWindow)
239 time.sleep(0.3)
240
241 # Envio de mensagens pelo WhatsApp Web
242 search_box = navegador.find_element('xpath', '//*[@id="side"]/div[1]/div/div/div
[2]/div/div[2]')
243
244 # Iterao para enviar mensagens para diferentes grupos
245 for x in range(11):
246     grupo = grupos[x]
247     tabela = consolidado[consolidado['Sala Rota'] == x]
248     tabela.reset_index(drop=True, inplace=True)
249     search_box.send_keys(grupo)
250     search_box.send_keys(Keys.ENTER)

```

```

251
252     # Iterao para enviar mensagens para diferentes clientes dentro do grupo
253     for i, cliente in enumerate(tabela['Cliente']):
254         # Extrair informaes do DataFrame para enviar na mensagem
255         veiculo = tabela.loc[i, 'Veiculo']
256         motorista = tabela.loc[i, 'Motorista']
257         supervisor = tabela.loc[i, 'Supervisor Rota']
258         valor = tabela.loc[i, 'Valor Devoluo (R$)']
259         pdv = tabela.loc[i, 'Cliente']
260         motivo = tabela.loc[i, 'Motivo Motorista']
261         rn = tabela.loc[i, 'Vendedor']
262         chegada = tabela.loc[i, 'Chegada ao PDV']
263         lacrado = tabela.loc[i, 'PDV Lacrado']
264         raio = tabela.loc[i, 'Acessou raio PDV?']
265         tempo = tabela.loc[i, 'Tempo no PDV']
266         volume = tabela.loc[i, 'Vol. HCL']
267
268         # Localizar caixa de texto e enviar mensagem com quebras de linha
269         text_box = navegador.find_element('xpath', '//*[@id="main"]/footer/div[1]/
div/span[2]/div/div[2]/div[1]/div/div[1]/p')
270         mensagem = ( f' - *{supervisor}*\n\n Cliente: {pdv}\n Setor: {rn}\n Nota
Fiscal: R${valor} | {volume}HCL\n Placa: {veiculo}\n Motorista: {motorista}\n\n *
Motivo: {motivo}*\n Chegou no PDV: {chegada}\n Tempo no PDV: {tempo}\n Acessou
Raio? {raio} | PDV lacrado?: {lacrado}')
271
272         # Enviar mensagem com quebras de linha
273         for one_line in mensagem.split("\n"):
274             text_box.send_keys(one_line)
275             text_box.send_keys(Keys.SHIFT + Keys.ENTER)
276
277         print(mensagem)
278         print('*REINICIANDO PROCESSO*')
279
280     # Remover arquivos
281     remover_arquivos()
282     time.sleep(90)
283
284     # Alternar para a janela do navegador principal
285     navegador.switch_to.window(newHandle)
286     time.sleep(0.5)

```

Código A.1 – Código Fonte do bot