

Apresentação e análise da adoção de Automação Robótica de Processos de código aberto em ambientes web corporativos

Eliezer Pedro Vicente Junior¹, Hermano Perrelli de Moura¹

¹Centro de Informática – Universidade Federal de Pernambuco (UFPE)

Recife – PE – Brasil

{epvj,hermano}@cin.ufpe.br

Resumo. *Este artigo apresenta uma estrutura de Automação Robótica de Processos (RPA) construída com ferramentas open source, como Python, Pandas, Selenium, GitLab e Jenkins. O propósito é compartilhar uma arquitetura robusta e escalável, que já foi implementada em duas empresas e funciona até o momento, além de discutir os impactos dessa solução na otimização de processos empresariais. A metodologia adotada aborda o desenvolvimento técnico da solução, detalha os principais fatores que promovem seu sucesso. Entre os resultados alcançados, destacam-se a redução de custos operacionais, o aumento da eficiência na execução de tarefas e a possibilidade de realocar equipes para atividades mais estratégicas. O artigo também explora os desafios enfrentados durante o processo, como a resistência inicial às mudanças e ajustes técnicos necessários, situando essas questões em um contexto de melhoria contínua. Conclui-se que a solução baseada em ferramentas open source é uma alternativa viável e eficaz, com potencial para transformar os processos empresariais, promovendo maior agilidade, flexibilidade e retorno financeiro.*

Palavras-chave: Automação Robótica de Processos (RPA); Open source; Eficiência operacional; Escalabilidade; Otimização de processos

Abstract. *This article presents a Robotic Process Automation (RPA) framework built with open-source tools such as Python, Pandas, Selenium, GitLab, and Jenkins. The purpose is to share a robust and scalable architecture that has already been implemented in two companies and is currently operational, as well as to discuss the impact of this solution on business process optimization. The adopted methodology addresses the technical development of the solution and details the key factors promoting its success. Among the achieved results, highlights include reduced operational costs, increased task execution efficiency, and the ability to reallocate teams to more strategic activities. The article also explores the challenges faced during the process, such as initial resistance to change and necessary technical adjustments, situating these issues within a continuous improvement context. It concludes that the solution based on open-source tools is a viable and effective alternative, with the potential to transform business processes by promoting greater agility, flexibility, and financial returns.*

Keywords: Robotic Process Automation (RPA); Open source; Operational efficiency; Scalability; Process optimization.

1. Introdução

A Automação Robótica de Processos (RPA) vem se consolidando como uma solução estratégica para aumentar a eficiência operacional em empresas de diferentes segmentos. A RPA é uma tecnologia que utiliza software para automatizar tarefas repetitivas, baseadas em regras e com pouca ou nenhuma interação humana. Os robôs de software (ou "bots") imitam as ações humanas em interfaces de sistemas, como clicar em botões, preencher formulários ou copiar e colar informações. O objetivo do RPA é aumentar a eficiência, reduzir erros e liberar os funcionários de atividades operacionais, permitindo que eles se concentrem em tarefas mais estratégicas e criativas (Murdoch, R. 2020). Dessa forma, ao automatizar essas atividades manuais, o RPA não apenas reduz o tempo e os custos associados, mas também eleva a precisão das operações e libera os colaboradores para se dedicarem a tarefas de maior valor agregado.

O trabalho teve como objetivo compartilhar uma estrutura de RPA desenvolvida exclusivamente com ferramentas open source, como Python, Pandas, Selenium, GitLab e Jenkins, destacando sua aplicação prática e os benefícios que ela pode proporcionar. A arquitetura apresentada foi implantada com sucesso em duas empresas, demonstrando sua escalabilidade, flexibilidade e capacidade de atender a diferentes cenários e demandas empresariais, as quais foram avaliadas através dos casos.

Além de detalhar a solução técnica, o trabalho explora o impacto da automação nos processos organizacionais, com foco em resultados como a redução de custos operacionais e a otimização do tempo de execução de tarefas, os quais foram medidos através do resultado dos casos. O referencial teórico fornecido oferece uma visão abrangente sobre o conceito de RPA e as tecnologias utilizadas, estabelecendo uma base sólida para a análise apresentada.

A metodologia adotada neste estudo estrutura-se em três etapas. Primeiramente, será feita a contextualização teórica e técnica, explicando os fundamentos do RPA e das tecnologias empregadas, como Python, Pandas, Selenium, GitLab e Jenkins, para proporcionar uma compreensão aprofundada do funcionamento da solução. Em seguida, na segunda etapa, será documentada a arquitetura técnica da solução, abordando a implementação da estrutura de RPA, a integração das ferramentas utilizadas e a descrição do fluxo de CI/CD, orquestração e escalabilidade. Por fim, na terceira etapa, serão relatados quatro casos de implantação da arquitetura em duas empresas, documentando as adaptações realizadas, desafios enfrentados e os impactos observados na operação, como aumento de eficiência e otimização de processos. Até o momento, os casos apresentados demonstram que a estrutura proposta funciona conforme esperado, atendendo às necessidades das empresas e proporcionando os benefícios esperados.

Por meio dessa abordagem metodológica bem estruturada, este artigo tem como objetivo apresentar a estrutura de RPA open source, demonstrando sua aplicação prática

e os benefícios que pode proporcionar. Dessa forma, pretende-se contribuir para a disseminação de soluções acessíveis, flexíveis e escaláveis, permitindo que organizações de diferentes setores aproveitem os benefícios da automação de forma estratégica e sustentável.

2. Referencial Teórico

Para entender a implementação apresentada, é fundamental estabelecer um referencial teórico que aborde os conceitos-chave e as tecnologias empregadas. A seguir, são descritos os principais conceitos relacionados ao RPA e as ferramentas que compõem a estrutura técnica do estudo.

2.1. Automação Robótica de Processos

A Automação Robótica de Processos (RPA) é uma tecnologia que utiliza software para automatizar tarefas repetitivas, baseadas em regras e com pouca ou nenhuma interação humana. Os robôs de software (ou "bots") imitam as ações humanas em interfaces de sistemas, como clicar em botões, preencher formulários ou copiar e colar informações. O objetivo do RPA é aumentar a eficiência, reduzir erros e liberar os funcionários de atividades operacionais, permitindo que eles se concentrem em tarefas mais estratégicas e criativas (Murdoch, R. 2020).

2.2. Python

Python é uma linguagem de programação de alto nível conhecida por sua simplicidade, legibilidade e grande ecossistema de bibliotecas. Por ser uma ferramenta extremamente flexível, Python é amplamente utilizado no desenvolvimento de soluções em RPA, permitindo desde a automação de tarefas simples até integrações complexas entre sistemas (Lutz, 2013).

2.3. Pandas

Pandas é uma biblioteca open source desenvolvida para manipulação e análise de dados em Python. Com estruturas de dados como os DataFrames, ela facilita o processamento de grandes volumes de informações de forma eficiente. Em pipelines de automação, Pandas é amplamente utilizada para organizar e transformar dados, proporcionando análises rápidas e estruturadas (McKinney, 2017).

2.4. Selenium

O Selenium é uma ferramenta open-source amplamente utilizada para automação de testes de aplicações web. Ela permite a simulação de interações humanas com páginas web, como cliques, preenchimento de formulários, navegação e execução de scripts JavaScript. O Selenium é uma das ferramentas mais populares para automação de testes devido à sua flexibilidade, suporte a múltiplos navegadores (Chrome, Firefox, Safari, etc.) e integração com diversos frameworks de teste, como JUnit, TestNG e Cucumber.

Ele é usado tanto para testar a funcionalidade de websites quanto para realizar automações mais complexas em ambientes de desenvolvimento e produção (Gundechea, U., 2013).

2.5. GitLab

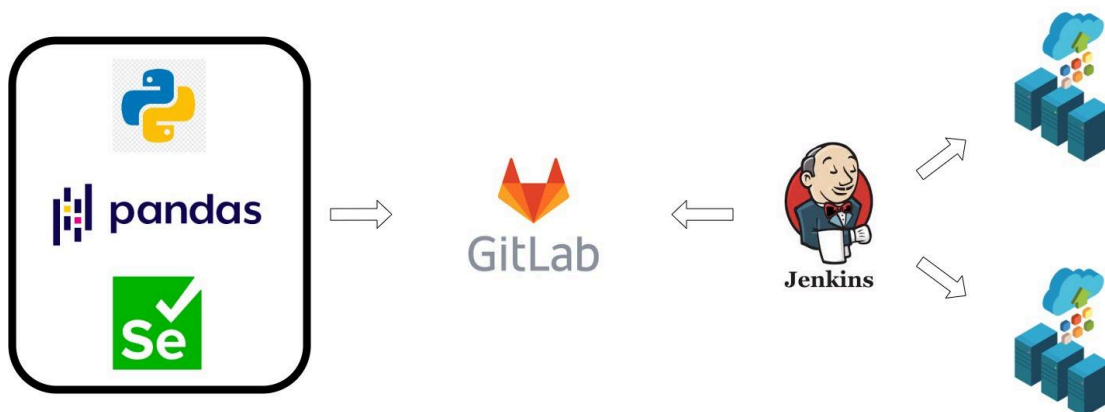
O GitLab é uma plataforma de DevOps baseada em Git que oferece uma série de ferramentas para automação de processos de desenvolvimento de software, incluindo controle de versão, integração contínua (CI), entrega contínua (CD) e monitoramento. GitLab permite que equipes de desenvolvimento colaborem de forma eficiente em projetos, gerenciando código-fonte, integrando testes automatizados, automatizando o processo de deployment e rastreando o progresso dos projetos. A plataforma também oferece recursos de revisão de código, gestão de problemas (issues) e segurança (Evertse, J., 2019).

2.6. Jenkins

Jenkins é uma ferramenta amplamente utilizada para integração e entrega contínua (CI/CD). Com sua capacidade de gerenciar pipelines de desenvolvimento e executar scripts com base em gatilhos ou agendamentos, Jenkins desempenha um papel crucial em arquiteturas de automação robustas e escaláveis (Smart, 2011).

3. Arquitetura e Fluxo de Automação

Imagem 2. Arquitetura da solução

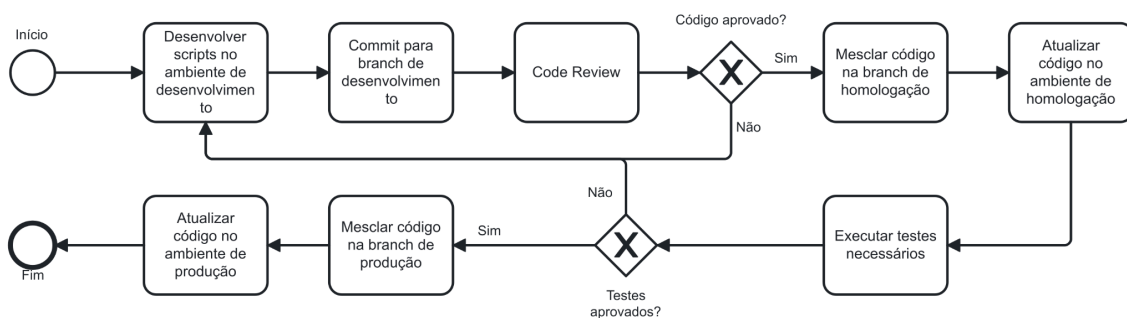


A arquitetura integrada garante que os robôs sejam projetados, testados e executados dentro de um fluxo automatizado consistente e escalável, proporcionando alta confiabilidade na automação e atendendo de forma eficiente às demandas específicas dos ambientes empresariais. São abordados aspectos como o desenvolvimento dos scripts em Python, o uso de Pandas e Selenium para manipulação de dados e interações web, o versionamento com GitLab, a orquestração e execução automatizada via Jenkins, além da implementação de um fluxo estruturado para garantir a confiabilidade e a escalabilidade das automações.

A solução funciona de forma estruturada para garantir um fluxo contínuo e automatizado de desenvolvimento, testes e implantação dos scripts. Os scripts são criados no ambiente de desenvolvimento, usando a linguagem de programação Python, a biblioteca Pandas e o framework Selenium. Após o desenvolvimento do script, um commit é feito para a branch de desenvolvimento no GitLab, onde o código será revisado através de um processo de code review. Se aprovado, o código é mesclado na branch de homologação; caso contrário, retorna ao desenvolvedor para ajustes necessários.

Assim que houver uma atualização na branch de homologação, um plugin do GitLab no Jenkins dispara um job que atualiza o ambiente de homologação com o novo código. Nesse ambiente, são realizados todos os testes necessários para garantir a qualidade e o funcionamento da aplicação. Caso os testes identifiquem falhas, o código volta para o desenvolvedor; caso não haja falhas, ele é mesclado na branch de produção. Da mesma forma, uma atualização na branch de produção aciona um job no Jenkins que atualiza o ambiente produtivo com a versão final do código.

Imagem 1. Fluxograma da solução



Dessa forma, esta seção detalha o ciclo completo de desenvolvimento, versionamento, execução e monitoramento dos scripts de automação, apresentando as ferramentas utilizadas e as melhores práticas adotadas para garantir a qualidade, segurança e eficiência dos processos automatizados.

3.1. Desenvolvimento dos Scripts de Automação

Os scripts são desenvolvidos e testados localmente ou em ambientes de desenvolvimento, garantindo que o robô execute as tarefas conforme esperado.

O Python é a base dos scripts de automação devido à sua simplicidade, ampla disponibilidade de bibliotecas e curva de aprendizado suave.

O Pandas é utilizado para manipulação e transformação de dados, permitindo tratar informações provenientes de fontes diversas, como APIs ou scraping de websites. A estrutura de dados em DataFrames facilita o processamento, a análise e a geração de relatórios.

O Selenium é empregado para a automação de interações com navegadores web, como login, preenchimento de formulários, download de relatórios e cliques em botões.

3.2. Versionamento e Controle de Código com GitLab

A branch de desenvolvimento é onde novas funcionalidades são criadas, garantindo que mudanças em teste não afetem o ambiente de produção.

As revisões de código são realizadas para assegurar a qualidade, segurança e aderência às boas práticas.

Após aprovação, o código é mesclado à branch principal (main), representando a versão estável pronta para produção.

3.3. Orquestração e Execução com Jenkins

A cada atualização na branch de produção, um job no Jenkins é acionado automaticamente, atualizando o ambiente de produção com o novo código.

Os RPAs são configurados para serem executados em horários específicos, como de madrugada, garantindo a realização de tarefas regulares, como geração de relatórios ou manutenção de sistemas.

Diferentes ambientes (desenvolvimento, homologação e produção) são gerenciados pelo Jenkins, que direciona a execução para o local adequado conforme as configurações definidas.

3.4. Integração Contínua (CI) e Entrega Contínua (CD)

Com cada atualização no código, o Jenkins executa testes automatizados para garantir que os scripts funcionem conforme o esperado antes de serem implantados. A Integração Contínua (CI) é uma prática de desenvolvimento que envolve a automação da integração de código de múltiplos desenvolvedores em um repositório compartilhado, garantindo que novas alterações sejam testadas continuamente para detectar erros rapidamente (Fowler, 2006). Já a Entrega Contínua (CD) expande essa abordagem ao automatizar a implantação do código aprovado em ambientes de produção, permitindo que novas funcionalidades e correções sejam entregues de forma rápida e confiável (Humble & Farley, 2010). Após a aprovação nos testes, o código atualizado é automaticamente implantado no ambiente de produção. Isso elimina a necessidade de intervenções manuais, acelerando o lançamento de novas funcionalidades ou correções e reduzindo o tempo necessário para entregas.

3.5. Monitoramento e Feedback Contínuo

O Jenkins acompanha as execuções e gera logs detalhados, permitindo a rápida identificação e resolução de problemas que possam surgir durante o processo. Além disso, um ciclo de monitoramento e feedback contínuo é mantido, possibilitando ajustes

nos scripts para se adaptarem a mudanças em websites ou formatos de dados. Isso garante que a automação permaneça eficiente e funcional.

4. Implantação e Resultados

O impacto da automação foi analisado em aspectos como a carga de trabalho, a motivação da equipe, os erros de digitação e o tempo gasto nas atividades. Para isso, foi utilizado um questionário detalhado, disponível no Apêndice A, enviado para stakeholders dos processos. O questionário foi respondido por quatro pessoas, todas líderes de equipe, garantindo uma visão estratégica sobre as mudanças trazidas pela automação.

As respostas foram coletadas via WhatsApp por meio de áudios, permitindo uma abordagem mais dinâmica e acessível. Posteriormente, esses áudios foram transcritos em um documento e consolidados neste texto, possibilitando uma análise mais estruturada do impacto da automação. O questionário comparou o cenário antes e depois da implantação dos robôs, destacando as melhorias observadas após a automação. O documento com a transcrição dos casos está disponível no GitHub e o link pode ser encontrado nas referências deste artigo.

4.1. Caso 1: Processamento de Notas Fiscais

O RPA recebe uma planilha com dados de notas fiscais e utiliza essas informações em um site para consultar e separá-las em duas categorias: DANFE e serviço. Em seguida, as notas são cadastradas no ERP da empresa.

4.1.1. Cenário antes da implantação

O processo manual de tratamento de notas fiscais era bastante demandante devido ao alto volume de atividades de diversas lojas da empresa. Com uma equipe de 35 pessoas, o fluxo envolvia consultar dados em um site, separar as notas fiscais em categorias (serviço ou DANFE) e cadastrá-las no ERP da empresa.

Essas tarefas eram demoradas e, mesmo com o esforço da equipe, frequentemente exigiam horas extras para atender às prioridades. O caráter repetitivo do trabalho gerava acúmulo de pendências e, embora os erros de digitação não fossem extremamente frequentes, sua recorrência impactava a qualidade do trabalho.

4.1.2. Processo de implantação

A introdução do robô trouxe transformações significativas na rotina. Inicialmente, a mudança enfrentou alguns desafios: o fluxo de trabalho precisou ser ajustado e a aceitação por parte da equipe foi mista. Enquanto alguns colaboradores receberam a automação com entusiasmo, outros mostraram resistência, temendo possíveis impactos nos seus empregos.

A implantação começou de forma gradual, sendo testada em uma única loja. O robô foi configurado para operar durante a madrugada, encarregando-se das tarefas mais

demoradas, como a separação e o cadastro de notas no ERP. Após ajustes técnicos e organizacionais, a aceitação geral da equipe melhorou, tornando o processo mais fluido.

4.1.3. Resultados após a implantação

A automação trouxe impactos positivos significativos. Com o robô assumindo tarefas repetitivas, os analistas passaram a se concentrar em atividades que exigem maior análise técnica. Isso não apenas reduziu a sobrecarga de trabalho como também melhorou a motivação da equipe.

O número de colaboradores dedicados ao processo foi reduzido de 35 para 17 pessoas, sem prejuízo da eficiência. Os demais foram realocados para áreas mais estratégicas. O robô também eliminou erros de digitação, já que o cadastro no ERP passou a ser feito com total precisão. No entanto, como os erros de digitação não eram tão frequentes antes da automação, então não foi possível medir os erros anteriormente.

Além disso, o tempo de execução foi drasticamente reduzido. O trabalho que antes levava de dois a três dias para ser concluído manualmente passou a ser realizado durante a madrugada, com as notas do dia anterior já processadas antes do início do expediente. Isso otimizou o fluxo de trabalho diário, aumentando a agilidade e a eficiência do processo. Não foi possível medir com precisão o tempo gasto anteriormente em horas, pois a demanda variava conforme a prioridade e a quantidade de notas a serem processadas.

4.2. Caso 2: Geração de Borderôs de Pagamento

Automação responsável por gerar um relatório e a partir deste, processar informações de pagamento e gerar borderôs de pagamentos em formato PDF e enviar por email.

4.2.1. Cenário antes da implantação

O processo manual de geração do borderô em PDF exigia um esforço considerável da equipe, embora a demanda não fosse excessivamente alta. Com uma equipe de quatro pessoas, cada colaborador era responsável pelo processamento das informações de uma marca, o que tornava a execução da tarefa um fator limitante na produtividade geral.

Apesar do controle sobre os prazos, a atividade consumia um tempo significativo da equipe, dificultando a alocação de esforços para outras tarefas analíticas e estratégicas. Embora os erros de digitação não fossem recorrentes, ocorrências pontuais impactavam a precisão das informações processadas.

4.2.2. Processo de implantação

A introdução do robô foi bem recebida pela equipe, que rapidamente sincronizou seu fluxo de trabalho com a automação. A aceitação foi ágil, especialmente após os primeiros testes, que demonstraram ganhos imediatos de eficiência. A automação eliminou a necessidade de gerar relatórios e processar dados manualmente, entregando

diretamente o arquivo PDF no e-mail da equipe. Esse avanço otimizou o fluxo de trabalho e proporcionou maior confiabilidade nas informações geradas.

4.2.3. Resultados após a implantação

A automação trouxe impactos significativos na rotina da equipe. Com a redução do tempo dedicado à atividade operacional, os colaboradores puderam realocar esforços para tarefas que exigiam maior análise e tomada de decisão. O tempo de processamento, que anteriormente exigia cerca de 8 horas diárias para quatro marcas, foi reduzido para um intervalo entre 30 minutos e 1 hora.

Além da redução no tempo, houve também uma otimização na alocação de pessoal. Antes da automação, quatro pessoas eram necessárias para lidar com as demandas das marcas; após a implantação do RPA, uma única pessoa passou a ser suficiente para supervisionar o processo e realizar as análises necessárias.

Outro benefício foi a eliminação de erros de digitação, garantindo maior precisão no processamento das informações. A integração do robô ao fluxo de trabalho permitiu que os colaboradores focassem em atividades estratégicas, aumentando a produtividade e a eficiência operacional do setor.

4.3. Caso 3: Lançamentos Financeiros no ERP

Robô responsável pelo download de um relatório financeiro no site da montadora, filtrar esse relatório de acordo com algumas regras definidas e fazer o lançamento dos dados financeiros no ERP da empresa.

4.3.1. Cenário antes da implantação

Antes da implementação do RPA, a equipe lidava diariamente com um grande volume de dados a serem processados e inseridos no sistema. Apesar da alta demanda, o time conseguia atender às solicitações na maioria das vezes, raramente ultrapassando os prazos estabelecidos.

A equipe era composta, em média, por cinco pessoas, que se dividiam para garantir que todas as lojas fossem atendidas. No entanto, a complexidade do processo e o grande número de informações a serem manipuladas tornavam a atividade desgastante. Além disso, erros de digitação eram frequentes, impactando a precisão dos dados registrados.

4.3.2. Processo de implantação

A implementação do RPA trouxe desafios iniciais, pois o processo possuía diversas variáveis que exigiam respostas específicas do robô para cada situação. Foi necessário um período de ajustes e refinamento dos filtros para garantir que o sistema funcionasse corretamente em todos os cenários possíveis.

Apesar dessas dificuldades iniciais, após a adaptação do RPA às particularidades do processo, a equipe rapidamente aceitou a automação. A melhora no desempenho e na motivação do time já pôde ser percebida nessa fase, uma vez que a automação reduziu significativamente o tempo gasto com atividades repetitivas.

4.3.3. Resultados após a implantação

Com a automação em funcionamento, o impacto positivo foi evidente. A atividade, que antes demandava muito tempo da equipe, passou a ser gerenciada com muito mais eficiência. Como resultado, o número de colaboradores necessários para a execução do processo foi reduzido de cinco para apenas duas pessoas, sendo uma responsável por cada marca.

Além da otimização da equipe, os erros de digitação foram praticamente eliminados, garantindo maior precisão nos dados inseridos no sistema. O tempo antes gasto com tarefas operacionais pôde ser redirecionado para atividades estratégicas, aumentando a produtividade do setor.

Outro benefício significativo foi a redução do tempo de execução do processo. Antes da automação, cada loja demandava entre 1 e 3 horas para conclusão das atividades, resultando em uma carga mínima de 20 horas diárias para toda a operação. Com o RPA, o robô inicia suas operações às 9h da manhã e conclui todas as tarefas antes do meio-dia, levando cerca de 3 horas para processar os dados de todas as lojas. Essa otimização garantiu maior agilidade e eficiência operacional, liberando tempo da equipe para outras atividades de maior valor agregado.

4.4. Caso 4: Baixa de Pedidos Pendentes de Recebimento

Robô responsável por dar baixa nos pedidos pendentes de recebimento. O RPA faz o download de dois relatórios, compara esses relatórios seguindo algumas regras definidas e registra no ERP os dados obtidos

4.4.1. Cenário antes da implantação

O processo manual de tratamento de dados era demorado, exigindo um esforço significativo da equipe. Com um time de aproximadamente quatro pessoas, a atividade envolvia a análise e inserção de informações no sistema, o que demandava bastante tempo e atenção.

Apesar de a demanda não ser extremamente alta, o processo manual tornava a execução lenta, impactando a produtividade geral. A equipe conseguia manter os prazos, mas o tempo investido na atividade impedia que os colaboradores se dedicassem a tarefas mais estratégicas. Além disso, erros de digitação, embora não frequentes, ocorriam ocasionalmente, exigindo revisões e correções.

4.4.2. Processo de implantação

A introdução do RPA trouxe desafios iniciais. Como o processo envolvia diversas variáveis, a automação precisou passar por múltiplos ajustes e validações para garantir que todas as situações fossem tratadas corretamente. Isso exigiu um período de testes e refinamento, mas, com o tempo, a equipe conseguiu alinhar o fluxo de trabalho com a nova ferramenta.

Mesmo com as dificuldades iniciais, a aceitação da equipe foi positiva. Após a finalização dos ajustes, o RPA passou a operar conforme o esperado, e a equipe se adaptou rapidamente ao novo modelo de trabalho.

4.4.3. Resultados após a implantação

Com a automação, os colaboradores puderam redirecionar seus esforços para atividades que exigem maior análise e tomada de decisão. O tempo anteriormente gasto no processo manual foi reduzido drasticamente, aumentando a eficiência do time.

Antes da automação, cerca de quatro pessoas eram necessárias para a execução das tarefas. Com a implantação do RPA, apenas um colaborador passou a ser responsável pelo monitoramento do processo, garantindo que tudo ocorresse conforme esperado.

Além disso, a automação eliminou os erros de digitação e garantiu maior precisão no cadastro de dados. O tempo de execução também foi otimizado. O trabalho, que anteriormente levava entre 7 e 14 horas para ser concluído manualmente, agora é realizado em cerca de uma hora, com cada marca processada em menos de 10 minutos. Isso trouxe maior agilidade ao fluxo de trabalho, permitindo que a equipe se concentrasse em demandas mais estratégicas.

4.5. Discussão Geral

Os quatro casos apresentados demonstram como a automação de processos repetitivos e operacionais pode trazer benefícios significativos para empresas de diferentes segmentos. A introdução do RPA permitiu otimizar tarefas que antes exigiam grande esforço humano, reduzindo o tempo de execução, minimizando erros e permitindo que os colaboradores se concentrem em atividades de maior valor estratégico.

Em todos os casos, houve uma redução considerável no número de colaboradores necessários para a execução dos processos, sem comprometer a eficiência. Pelo contrário, a automação garantiu maior precisão nas operações e acelerou o fluxo de trabalho. Além disso, a aceitação das equipes variou entre resistência inicial e rápida adaptação, demonstrando que a mudança para processos automatizados pode ser desafiadora, mas, quando bem implementada, gera impactos positivos.

Outro fator comum foi a necessidade de ajustes e refinamentos nos primeiros momentos de implantação. Isso reforça a importância de um período de testes e adaptação para garantir que a automação funcione corretamente e atenda às necessidades específicas de cada empresa.

Os quatro casos continuam operando de forma eficiente até o momento, evidenciando que a estrutura proposta de RPA open source é funcional, escalável e flexível o suficiente para diferentes cenários.

Tabela 1. Impactos dos casos estudados

Casos	Nº de Colaboradores Antes	Nº de Colaboradores Depois	Tempo Antes	Tempo Depois	Redução de Erros	Outros Benefícios
1 Processamento de Notas Fiscais	35	17	2 a 3 dias	Algumas horas (durante a madrugada)	Eliminados	Maior agilidade e eficiência no fluxo de trabalho
2 Geração de Borderô de Pagamento	4	1	8 horas diárias	30 minutos a 1 hora	Eliminados	Otimização da alocação de pessoal
3 Lançamentos Financeiros no ERP	5	2	20 horas diárias	3 horas para todas as lojas	Eliminados	Maior precisão e realocação da equipe para atividades estratégicas
4 Baixa de Pedidos Pendentes	4	1	7 a 14 horas	1 hora (cada marca processada em menos de 10 min)	Eliminados	Maior agilidade e liberação da equipe para outras atividades

5. Conclusão e Trabalhos Futuros

A adoção de uma estrutura de RPA baseada em ferramentas open source, como Python, Pandas, Selenium, GitLab e Jenkins, mostrou-se uma solução eficaz, escalável e

acessível para a automação de processos repetitivos em diferentes cenários. Os quatro casos analisados demonstram como a automação pode transformar a execução de tarefas operacionais, reduzindo o tempo de processamento, minimizando erros e permitindo que as equipes se concentrem em atividades de maior valor agregado.

No Caso 1, referente ao processamento de notas fiscais, um processo que exigia uma equipe de 35 pessoas e até três dias para ser concluído passou a ser realizado por 17 colaboradores em questão de horas, garantindo maior precisão e eficiência. O robô eliminou erros de digitação e permitiu que as notas fossem processadas automaticamente durante a madrugada, otimizando o fluxo de trabalho e aumentando a produtividade.

No Caso 2, relacionado à geração de borderôs de pagamento, o tempo de processamento, que antes demandava cerca de 8 horas diárias para quatro marcas, foi reduzido para um intervalo entre 30 minutos e 1 hora. Além disso, a equipe necessária para a tarefa foi reduzida de quatro para apenas uma pessoa, permitindo um melhor aproveitamento dos recursos humanos e eliminando erros manuais.

No Caso 3, sobre lançamentos financeiros no ERP, um processo que antes exigia cinco colaboradores e até 20 horas diárias para ser concluído foi reduzido para um tempo de execução de apenas três horas, com duas pessoas supervisionando a automação. A precisão dos dados também foi aprimorada, eliminando falhas de digitação e garantindo maior confiabilidade nas informações.

Por fim, no Caso 4, referente à baixa de pedidos pendentes de recebimento, a automação reduziu drasticamente o tempo de execução do processo, que anteriormente levava entre 7 e 14 horas para ser concluído manualmente e passou a ser finalizado em apenas uma hora. A equipe necessária para essa atividade foi reduzida de quatro para um colaborador, permitindo maior foco em atividades estratégicas.

Embora a implementação da automação tenha enfrentado desafios iniciais, como resistência à mudança, necessidade de ajustes no fluxo de trabalho e refinamento dos critérios de validação, esses obstáculos foram superados por meio de comunicação clara, treinamentos e ciclos contínuos de feedback. A flexibilidade e escalabilidade da arquitetura proposta foram fundamentais para o sucesso da automação, permitindo adaptações rápidas a diferentes processos e demandas. Além disso, a utilização de ferramentas de CI/CD, como o Jenkins, possibilitou atualizações constantes sem impactar a operação.

Diante dos resultados positivos observados, recomenda-se que empresas interessadas na implementação de RPA open source realizem um mapeamento detalhado dos processos a serem automatizados, invistam em treinamentos para as equipes e garantam uma infraestrutura tecnológica robusta. Além disso, é essencial estabelecer ciclos de revisão e manutenção contínua para assegurar que a automação acompanhe as mudanças nos fluxos de trabalho.

Os casos analisados evidenciam que a solução não apenas melhora a eficiência operacional, mas também gera impactos significativos na produtividade e motivação das equipes. Essa abordagem proporciona maior agilidade, precisão e retornos financeiros expressivos, com custos acessíveis e alta escalabilidade, tornando-se uma alternativa viável para empresas que buscam inovação e otimização operacional.

Trabalhos futuros podem se aprofundar no desenvolvimento de diretrizes para a adoção e aprimoramento da automação baseada em RPA open source no ambiente corporativo, abordando desde a identificação de processos com alto potencial de automação até a definição de métricas para avaliação contínua dos benefícios gerados. Além disso, pesquisas podem explorar estratégias para integração eficiente com diferentes sistemas legados, garantindo compatibilidade e escalabilidade das soluções automatizadas.

Referências

MURDOCH, R. *Robotic process automation: Guide to building software robots, automate repetitive tasks, and become an RPA consultant*. Packt Publishing, 2020.

LUTZ, M. *Learning Python*. O'Reilly Media, 2013.

MCKINNEY, W. *Python for Data Analysis*. O'Reilly Media, 2017.

GUNDECHA, U. *Selenium testing tools cookbook*. Packt Publishing, 2013.

EVERTSE, J. *Mastering GitLab 12: Implement DevOps culture and repository management with GitLab*. Packt Publishing, 2019.

SMART, J. *Jenkins: The Definitive Guide*. O'Reilly Media, 2011.

FOWLER, M. *Continuous Integration*. Martin Fowler, 2006. Disponível em: <https://martinfowler.com/articles/continuousIntegration.html>. Acesso em: 16 mar. 2025.

HUMBLE, J.; FARLEY, D. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Addison-Wesley, 2010.

PEDRO, Eliezer. Transcrição dos cases. 2025. Disponível em: <https://github.com/eliezerpedro/TCC-RPA/blob/main/Transcri%C3%A7%C3%A3o%20dos%20cases.pdf>. Acesso em: 3 abr. 2025.

Apêndice A: Questionário Utilizado na Pesquisa

Este questionário foi utilizado como instrumento de coleta de dados para compreender o impacto da implementação dos robôs na rotina operacional das empresas. As perguntas estão organizadas em tópicos que abordam o cenário antes, durante e após a implantação da solução de RPA.

1. Antes da Implantação dos Robôs

- Como era o dia a dia antes da implantação dos robôs?
 - Havia muita demanda e era difícil dar conta?
 - As entregas eram feitas fora do prazo?
 - Quantas pessoas (em média) compunham a equipe?
 - Quais eram as maiores dificuldades enfrentadas?
 - Erros de digitação eram frequentes?

2. Processo de Implantação dos Robôs

- O processo de implantação foi complicado?
- Como a equipe reagiu à nova mudança?
- A aceitação foi rápida?
- Houve melhora no desempenho e na motivação da equipe já na fase de implantação?

3. Após a Implantação dos Robôs

- Como passou a ser o dia a dia após a implantação?
- O desempenho e a motivação da equipe melhoraram?
- Os problemas persistiram mesmo após a implantação?
- Quantas pessoas (em média) eram necessárias para realizar as atividades anteriormente, e quantas foram necessárias após a implantação dos robôs?
- Os erros de digitação reduziram?

4. Comparação de Tempo com e sem os Robôs

- Quanto tempo (em média) a equipe levava para realizar as atividades antes da implantação dos robôs?
- Quanto tempo (em média) passou a ser necessário para realizar as mesmas atividades com os robôs?