v7.n9.2026

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

MODELAGEM E MELHORIA DE PROCESSOS DE NEGÓCIO COM BPMN E AUTOMAÇÃO EM PYTHON: UM ESTUDO DE CASO NA GESTÃO DE DADOS

BUSINESS PROCESS MODELING AND IMPROVEMENT WITH BPMN AND PYTHON AUTOMATION: A CASE STUDY IN DATA MANAGEMENT

MODELADO Y MEJORA DE PROCESOS DE NEGOCIO CON BPMN Y AUTOMATIZACIÓN EN PYTHON: UN ESTUDIO DE CASO EN LA GESTIÓN DE DATOS

Paulo Esteves da Cunha Junior[1]; Felipe Diniz Dallilo[2]

e799077
https://doi.org/10.47820/recima21.v7i9.9077

PUBLICADO: 09/2026

RESUMO

Este trabalho aborda a análise e a melhoria de processos de negócio por meio da aplicação de *Business Process Management* (BPM), *Business Process Model and Notation* (BPMN) e automação com Python. Parte-se do pressuposto de que atividades manuais, repetitivas e baseadas em regras podem comprometer a eficiência operacional, ampliar a ocorrência de erros e elevar o tempo de execução dos processos. O estudo tem como objeto o processo de Revisão da Linha de Base de Custos, desenvolvido no Gerenciamento de Custos do *Project Management Office* (PMO), com foco na comparação entre o orçamento da linha de base e as revisões trimestrais. A pesquisa caracteriza-se como aplicada, exploratória e descritiva, com abordagem qualitativa e quantitativa, desenvolvida por meio de revisão bibliográfica e estudo de caso. Nesse contexto, o processo foi modelado no estado atual (*AS-IS*), possibilitando a identificação de uma lacuna operacional e a elaboração de um modelo futuro (*TO-BE*), apoiado pela automação de uma atividade específica por meio de Python. Os resultados demonstraram redução do *lead time* total do subprocesso de 25 para 13 dias, equivalente a 48%, eliminação do tempo associado às atividades de retrabalho, de 8 para 0 dias, e redução de 80% no tempo da atividade central de comparação, de 5 para 1 dia. Conclui-se que a integração entre BPM, BPMN e automação com Python contribuiu para simplificar e padronizar o fluxo, reduzir inconsistências e melhorar a eficiência operacional do subprocesso analisado.

PALAVRAS-CHAVE: Automação de processos. BPM. BPMN. Python. Gestão de dados.

ABSTRACT

This study addresses the analysis and improvement of business processes through the application of Business Process Management (BPM), Business Process Model and Notation (BPMN), and automation with Python. It is based on the assumption that manual, repetitive, and rule-based activities may compromise operational efficiency, increase the occurrence of errors, and extend process execution time. The object of the study is the Cost Baseline Review process, carried out within Cost Management of the Project Management Office (PMO), focusing on the comparison between the baseline budget and quarterly reviews. The research is characterized as applied, exploratory, and descriptive, with a qualitative and quantitative approach, developed through a literature review and a case study. In this context, the process was modeled in its current state (AS-IS), enabling the identification of an operational gap and the development of a future-state model (TO-BE), supported by the automation of a specific activity using Python.

[1] Universidade de Araraquara, Graduando em Sistemas de Informação.
[2] Universidade de São Paulo, Doutor em Engenharia de Software.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

The results demonstrated a reduction in the subprocess total lead time from 25 to 13 days, equivalent to 48%, elimination of the time associated with rework activities, from 8 to 0 days, and an 80% reduction in the time of the central comparison activity, from 5 to 1 day. It is concluded that the integration of BPM, BPMN, and Python automation contributed to simplifying and standardizing the workflow, reducing inconsistencies, and improving the operational efficiency of the analyzed subprocess.

KEYWORDS: *Process automation. BPM. BPMN. Python. Data management.*

RESUMEN
Este trabajo aborda el análisis y la mejora de procesos de negocio mediante la aplicación de Business Process Management (BPM), Business Process Model and Notation (BPMN) y automatización con Python. Se parte del supuesto de que las actividades manuales, repetitivas y basadas en reglas pueden comprometer la eficiencia operativa, aumentar la ocurrencia de errores y prolongar el tiempo de ejecución de los procesos. El objeto de estudio es el proceso de Revisión de la Línea Base de Costos, desarrollado en la Gestión de Costos del Project Management Office (PMO), con énfasis en la comparación entre el presupuesto de la línea base y las revisiones trimestrales. La investigación se caracteriza como aplicada, exploratoria y descriptiva, con un enfoque cualitativo y cuantitativo, desarrollada mediante revisión bibliográfica y estudio de caso. En este contexto, el proceso fue modelado en su estado actual (AS-IS), lo que permitió identificar una brecha operativa y elaborar un modelo futuro (TO-BE), respaldado por la automatización de una actividad específica mediante Python. Los resultados demostraron una reducción del lead time total del subproceso de 25 a 13 días, equivalente al 48%, la eliminación del tiempo asociado a las actividades de retrabajo, de 8 a 0 días, y una reducción del 80% en el tiempo de la actividad central de comparación, de 5 a 1 día. Se concluye que la integración entre BPM, BPMN y la automatización con Python contribuyó a simplificar y estandarizar el flujo, reducir inconsistencias y mejorar la eficiencia operativa del subproceso analizado.

PALABRAS CLAVE: *Automatización de procesos. BPM. BPMN. Python. Gestión de datos.*

INTRODUÇÃO

Em um contexto organizacional marcado pela competitividade, pelo avanço tecnológico e pela exigência de respostas mais ágeis, precisas e padronizadas, a melhoria contínua dos processos de negócio passou a ocupar posição estratégica nas instituições. Nesse cenário, a gestão orientada por processos favorece a compreensão integrada das práticas organizacionais e o alinhamento entre operações, objetivos institucionais e geração de valor (GUIMARÃES; BRAGA; ZAIDAN, 2021).

Sob essa perspectiva, a execução manual de rotinas repetitivas, estruturadas e baseadas em regras tende a ampliar atrasos, retrabalho, inconsistências e falhas operacionais, comprometendo a padronização e a confiabilidade dos resultados. Nesse sentido, o *Business Process Management* (BPM) ou Gestão de Processos de Negócio destaca-se como abordagem

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

voltada à compreensão, análise, modelagem, execução, monitoramento e melhoria contínua dos fluxos organizacionais, favorecendo a identificação de gargalos, redundâncias e oportunidades de aperfeiçoamento (GUIMARÃES; BRAGA; ZAIDAN, 2021; DALLILO, 2025).

O *Business Process Model and Notation* (BPMN) ou Modelo e Notação de Processos de Negócio possibilita a modelagem estruturada de atividades, eventos, decisões, responsáveis e sequências de execução. Sua utilização contribui para documentar e interpretar analiticamente as rotinas, permitindo visualizar etapas manuais, pontos de espera, retrabalhos e tarefas passíveis de otimização ou automatização (DALLILO, 2025).

Paralelamente, a automatização de rotinas também vem se consolidando como estratégia relevante no contexto da transformação digital, pois pode reduzir esforço manual, ampliar a padronização, minimizar falhas operacionais e elevar a eficiência. Contudo, seus resultados tendem a ser mais consistentes quando decorrem de análise estruturada do fluxo sobre o qual incidem e não de iniciativa isolada (LIEVANO-MARTÍNEZ et al., 2022).

A partir dessa compreensão do fluxo, entre as tecnologias aplicáveis a esse tipo de solução, Python destaca-se por sua legibilidade, versatilidade e ampla disponibilidade de bibliotecas voltadas à manipulação de dados, integração entre sistemas e execução automatizada de tarefas. Tais características favorecem o desenvolvimento de soluções práticas e flexíveis, especialmente em rotinas estruturadas e recorrentes (TIERNEY, 2025).

À luz dessas considerações, este trabalho insere-se no campo da análise e melhoria de processos de negócio por meio da articulação entre BPM, BPMN e Python, tomando como objeto de estudo o processo de Revisão da Linha de Base de Custos desenvolvido na área de Gerenciamento de Custos do *Project Management Office* (PMO) ou Escritório de Projetos. De forma específica, o estudo concentra-se na etapa de comparação do orçamento da linha de base com as revisões trimestrais, na qual a comparação, a conferência e a consolidação de informações demandam elevado esforço manual e favorecem retrabalho, inconsistências e limitações operacionais.

A questão de pesquisa a ser investigada é: de que forma a modelagem do processo de Revisão da Linha de Base de Custos, fundamentada nos princípios do BPM e representada por meio da BPMN, associada à automatização com Python, pode contribuir para a redução de falhas, retrabalho e tempo de execução no âmbito do Gerenciamento de Custos do PMO?

A relevância da pesquisa manifesta-se, no plano acadêmico, pela contribuição às discussões sobre a integração entre gestão por processos, modelagem e automatização, ao reunir fundamentação teórica e aplicação concreta em uma mesma investigação (GUIMARÃES; BRAGA; ZAIDAN, 2021; DALLILO, 2025; LIEVANO-MARTÍNEZ et al., 2022). No plano prático,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

decorre da proposição de uma solução aplicada a uma rotina real do Gerenciamento de Custos do PMO com potencial para reduzir tempo de execução, retrabalho e erros operacionais, além de ampliar a padronização e a consistência no tratamento dos dados.

O objetivo geral deste trabalho é analisar o processo de Revisão da Linha de Base de Custos, modelá-lo com BPMN e propor sua melhoria por meio da automatização de uma etapa com Python no âmbito do Gerenciamento de Custos do PMO. Os objetivos específicos são:

- mapear o fluxo atual por meio do modelo *AS-IS*;
- identificar gargalos, falhas e tarefas repetitivas passíveis de automatização;
- propor um modelo *TO-BE* com aprimoramento do fluxo de trabalho;
- desenvolver um *script* em Python para automatizar a etapa selecionada; e
- comparar o desempenho antes e depois da melhoria proposta.

1. REFERENCIAL TEÓRICO

Esta seção articula os fundamentos da gestão e da modelagem de processos com a automatização em Python, buscando relacionar as contribuições dos autores à análise, ao redesenho e ao aperfeiçoamento do subprocesso investigado.

1.1. *Business Process Management* (BPM)

Guimarães, Braga e Zaidan (2021) e Dallilo (2025) convergem ao tratar o *Business Process Management* (BPM) como uma abordagem orientada à compreensão e à melhoria integrada dos processos. Enquanto Guimarães, Braga e Zaidan (2021) enfatizam o alinhamento entre processos, riscos e desempenho organizacional, Dallilo (2025) reforça a necessidade de estruturar e representar o fluxo para torná-lo analisável e passível de aperfeiçoamento.

Aganette (2021) complementa essa perspectiva ao relacionar a adoção do BPM a fatores organizacionais que impulsionam mudanças e melhoria contínua. Em conjunto, essas contribuições sustentam o uso dos modelos *AS-IS* e *TO-BE* como instrumentos para diagnosticar o estado atual, explicitar gargalos e organizar a proposição de um estado futuro mais eficiente.

No presente estudo, essa combinação teórica orienta o mapeamento do processo de Revisão da Linha de Base de Custos, a identificação de pontos críticos e a comparação entre o fluxo atual e o fluxo redesenhado.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

1.2. *Business Process Model and Notation* (BPMN)

No campo da representação de processos, Dallilo (2025) destaca a *Business Process Model and Notation* (BPMN) como notação capaz de tornar visíveis atividades, decisões, responsáveis e interações. Essa função de representação é particularmente útil quando a análise exige comparar o funcionamento atual com uma configuração futura e comunicar as mudanças de forma compreensível às áreas de negócio e tecnologia.

Amarilla e Iarozinski Neto (2018) aproximam a discussão da análise comparativa de processos de negócio, enquanto Araújo e Gomes (2022) tratam do gerenciamento de processos apoiado por ferramentas específicas. Essas perspectivas complementam a visão de Dallilo (2025), pois mostram que a modelagem não se limita à documentação: ela também apoia a comparação, o diagnóstico e a escolha de melhorias no fluxo.

Neste trabalho, a BPMN é utilizada para representar os modelos *AS-IS* e *TO-BE* e para evidenciar onde a automação altera a sequência das atividades, reduzindo retrabalho e tempo de execução.

1.3. Automatização de processos

A automatização de processos busca transferir para recursos computacionais tarefas repetitivas, estruturadas e baseadas em regras. Lievano-Martínez et al. (2022) associam a automação a ganhos de eficiência em processos produtivos, enquanto Guedes (2022) aproxima diretamente BPM e automatização com Python. Embora aplicados a contextos distintos, os estudos convergem ao indicar que a automação produz resultados mais consistentes quando parte de um processo previamente compreendido e estruturado.

Essa convergência é relevante para diferenciar a automação proposta de uma substituição pontual do trabalho manual. O objetivo não é apenas executar a mesma atividade mais rapidamente, mas incorporá-la ao redesenho do fluxo, de modo que a solução tecnológica reduza a necessidade de validações e correções posteriores.

No subprocesso analisado, essa lógica direciona a automação para a comparação entre a linha de base e as revisões trimestrais, etapa caracterizada por grande volume de dados, regras determinísticas e recorrência de conferências manuais.

1.4. Python como tecnologia para automatização

Python apresenta características favoráveis à automatização de tarefas estruturadas, especialmente em rotinas que envolvem dados tabulares, planilhas, validações e regras de negócio. Tierney (2025) demonstra seu uso na automação de avaliações computacionais,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

enquanto Bogni et al. (2022) exploram algoritmos voltados ao tratamento de indicadores e dados educacionais.

Cunha (2024), por sua vez, aplica Python à automatização de processos contábeis. Apesar das diferenças entre os domínios analisados, os estudos convergem quanto à capacidade da linguagem de reduzir esforço manual, organizar dados e executar rotinas repetitivas de forma reproduzível, o que sustenta sua adoção no presente estudo.

A solução proposta utiliza Python para ler planilhas, padronizar dados, comparar versões e gerar um relatório estruturado, empregando recursos como *pandas*, os e *openpyxl*. Sua efetividade é avaliada pela comparação entre os cenários *AS-IS* e *TO-BE* com base no *lead time* do subprocesso, no tempo associado a retrabalho e no desempenho da atividade central de comparação.

2. METODOLOGIA

A pesquisa caracteriza-se como aplicada, exploratória e descritiva, com abordagem qualitativa e quantitativa. A dimensão qualitativa concentra-se na análise do fluxo, de suas etapas, interações e gargalos, enquanto a quantitativa incide sobre a comparação dos resultados antes e depois da implementação da solução com base em indicadores relacionados ao tempo de execução, à ocorrência de falhas e ao retrabalho. Como procedimentos técnicos, adotam-se a revisão bibliográfica e o estudo de caso desenvolvido em uma empresa do setor aeronáutico, na área de PMO.

Para compreender a execução do subprocesso antes da melhoria proposta, analisou-se o processo de Revisão da Linha de Base de Custos, com foco na comparação entre o orçamento da linha de base e as revisões trimestrais. As informações levantadas subsidiaram a modelagem do estado atual (*AS-IS*) em BPMN, a partir da qual foram identificadas as lacunas e as oportunidades de melhoria do subprocesso. Com base nesse diagnóstico, elaborou-se o modelo futuro (*TO-BE*), com o objetivo de simplificar o fluxo e reduzir o *lead time* do subprocesso.

Depois da definição do processo *TO-BE*, foi desenvolvida uma solução em Python para auxiliar na comparação entre a revisão de base e a linha de base nesse subprocesso. A avaliação da solução foi realizada pela comparação entre os cenários anterior e posterior à implementação, considerando indicadores como o *lead time* estimado do subprocesso, o tempo associado a retrabalhos, a redução de atividades corretivas e o desempenho geral do fluxo.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

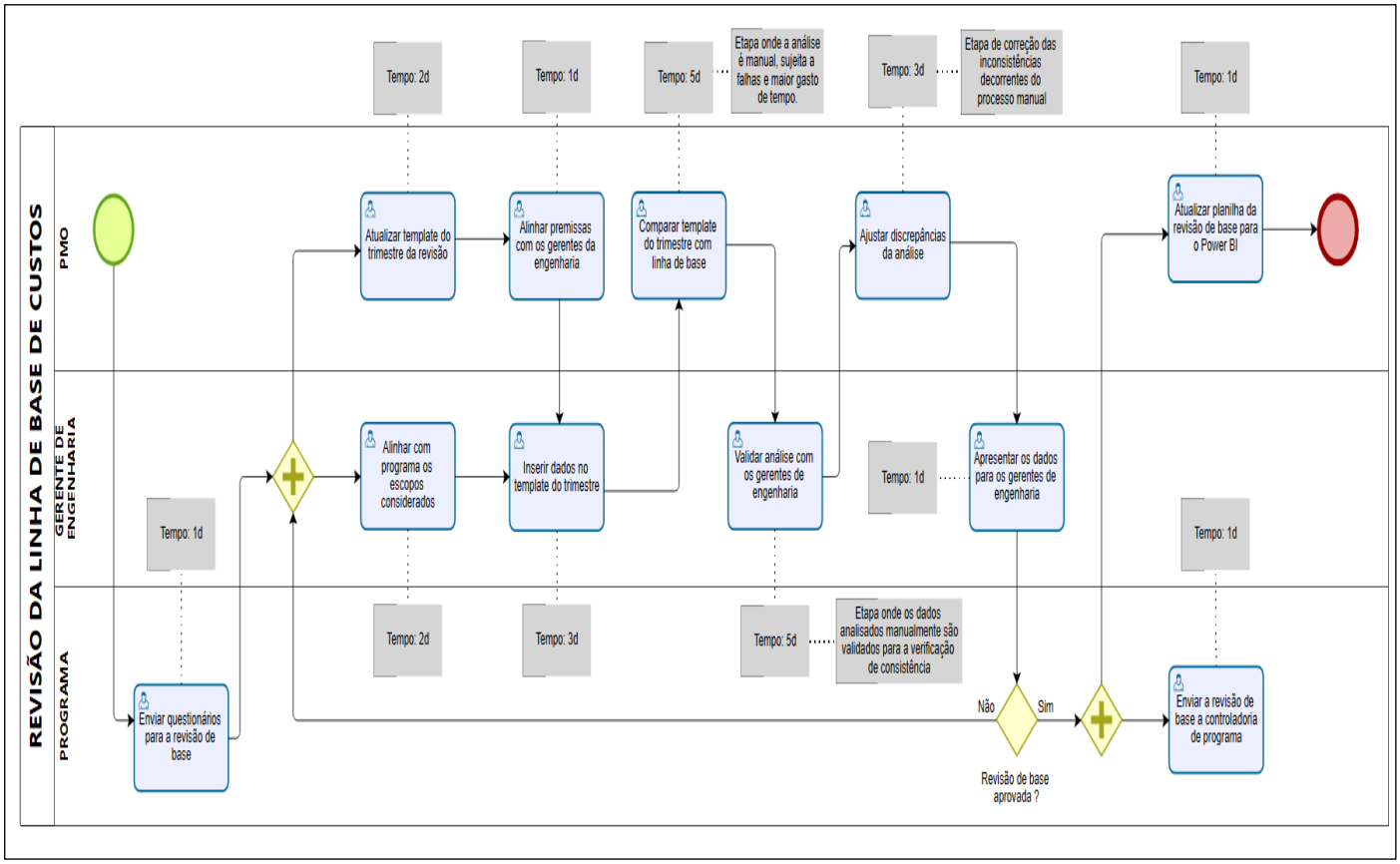
3. RESULTADOS E DISCUSSÃO

Esta seção apresenta o desenvolvimento da proposta aplicada ao subprocesso estudado, considerando a modelagem do estado atual (*AS-IS*), a identificação das lacunas existentes, a proposição do modelo futuro (*TO-BE*), o desenvolvimento da automação em Python e a análise dos resultados obtidos.

3.1. Modelagem do processo no estado atual (*AS-IS*)

Após a definição do escopo do estudo, buscou-se compreender como o subprocesso era executado antes da melhoria proposta. Para isso, foi analisado o processo de Revisão da Linha de Base de Custos em uma empresa do setor aeronáutico, na área de PMO, com foco na comparação entre o orçamento da linha de base e as revisões trimestrais. A partir das informações levantadas, foi elaborada a modelagem do estado atual (*AS-IS*) em BPMN, conforme apresentado na Figura 1.

Figura 1. Modelagem do subprocesso de Revisão da Linha de Base de Custos no estado atual (*AS-IS*)



Fonte: Autoria própria, 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

A Figura 1 representa o fluxo do subprocesso em seu estado atual, evidenciando a sequência das atividades, os responsáveis envolvidos e os principais pontos de interação entre as áreas participantes. Em termos funcionais, a modelagem mostra que o processo se inicia com o envio dos questionários para revisão de base, passa pelas etapas de atualização do *template*, alinhamento de escopos e premissas, inserção dos dados e comparação entre o *template* do trimestre e a linha de base, até chegar às fases de validação, ajuste, apresentação dos dados e envio da revisão à controladoria com posterior atualização para o Power BI.

Essa visualização permite compreender como o subprocesso está estruturado no cenário *AS-IS* e mostra de forma integrada onde se concentram as atividades manuais, os pontos de verificação e as etapas que mais influenciam o tempo total do fluxo.

A Tabela 1 permite visualizar as onze atividades que fazem parte do subprocesso analisado. Juntas, elas representam um *lead time* estimado de 25 dias.

Tabela 1. Distribuição do *lead time* por atividade do subprocesso no estado atual (*AS-IS*)

Atividade do Processo	**Código da Atividade**	***Lead time* Estimado (dias)**
Enviar questionários para a revisão de base	AT01	1
Atualizar *template* do trimestre da revisão	AT02	2
Alinhar com o programa os escopos considerados	AT03	2
Alinhar premissas com os gerentes da engenharia	AT04	1
Inserir dados no *template* do trimestre	AT05	3
Comparar *template* do trimestre com linha de base	AT06	5
Validar análise com os gerentes de engenharia	AT07	5
Ajustar discrepâncias da análise	AT08	3
Apresentar os dados para os gerentes de engenharia	AT09	1
Enviar a revisão de base à controladoria de programa	AT10	1
Atualizar planilha da revisão de base para o Power BI	AT11	1
Ciclo total do processo *AS-IS*	**CT *AS-IS***	**25**

Fonte: Autoria própria, 2026.

A análise da modelagem BPMN e dos tempos por atividade permitiu compreender como o subprocesso funciona no estado atual e serviu de base para a proposição do modelo futuro (*TO-BE*).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.2. Identificação de lacunas, gargalos e oportunidades de melhoria

Ao analisar o estado atual (*AS-IS*), foi possível identificar algumas lacunas no subprocesso de Revisão da Linha de Base de Custos. O levantamento mostrou que parte importante do tempo é consumida por tarefas manuais, validações e correções. No cenário atual, o subprocesso possui *lead time* estimado de 25 dias, com destaque para as atividades indicadas na Tabela 2.

Tabela 2. Atividades críticas do subprocesso no estado atual (*AS-IS*)

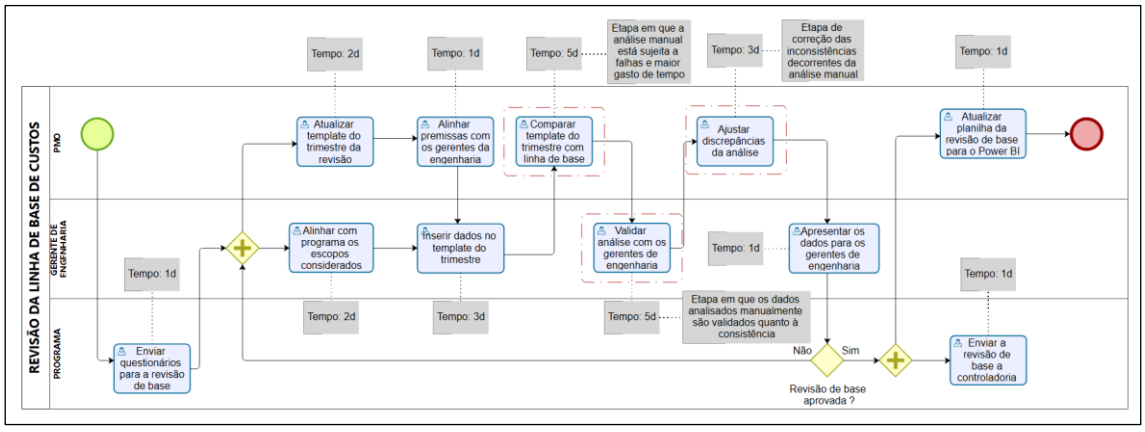
Atividade do Processo	Código da Atividade	*Lead time* Estimado (dias)
Comparar *template* do trimestre com linha de base	AT06	5
Validar análise com os gerentes de engenharia	AT07	5
Ajustar discrepâncias da análise	AT08	3
Ciclo total das atividades	**CTA**	**13**

Fonte: Autoria própria, 2026.

As três atividades analisadas somam 13 dias e correspondem a 52% do *lead time* total de 25 dias. Esse resultado evidencia que mais da metade do ciclo do subprocesso estava concentrada nas etapas de comparação, validação e correção das informações.

A AT06 constitui o ponto central desse gargalo, pois envolve a comparação manual entre o *template* do trimestre e a linha de base. Na sequência, as atividades AT07 e AT08 concentram validação e correção das discrepâncias, ampliando o esforço de conferência e o tempo de resposta do subprocesso.

Figura 2. Identificação de lacunas e pontos críticos no modelo *AS-IS* do subprocesso de Revisão da Linha de Base de Custos



Fonte: Autoria própria, 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

A Figura 2 destaca no modelo *AS-IS* as etapas consideradas mais críticas para o desempenho do subprocesso, com ênfase na comparação manual entre o *template* do trimestre e a linha de base, na validação da análise e no ajuste das discrepâncias identificadas.

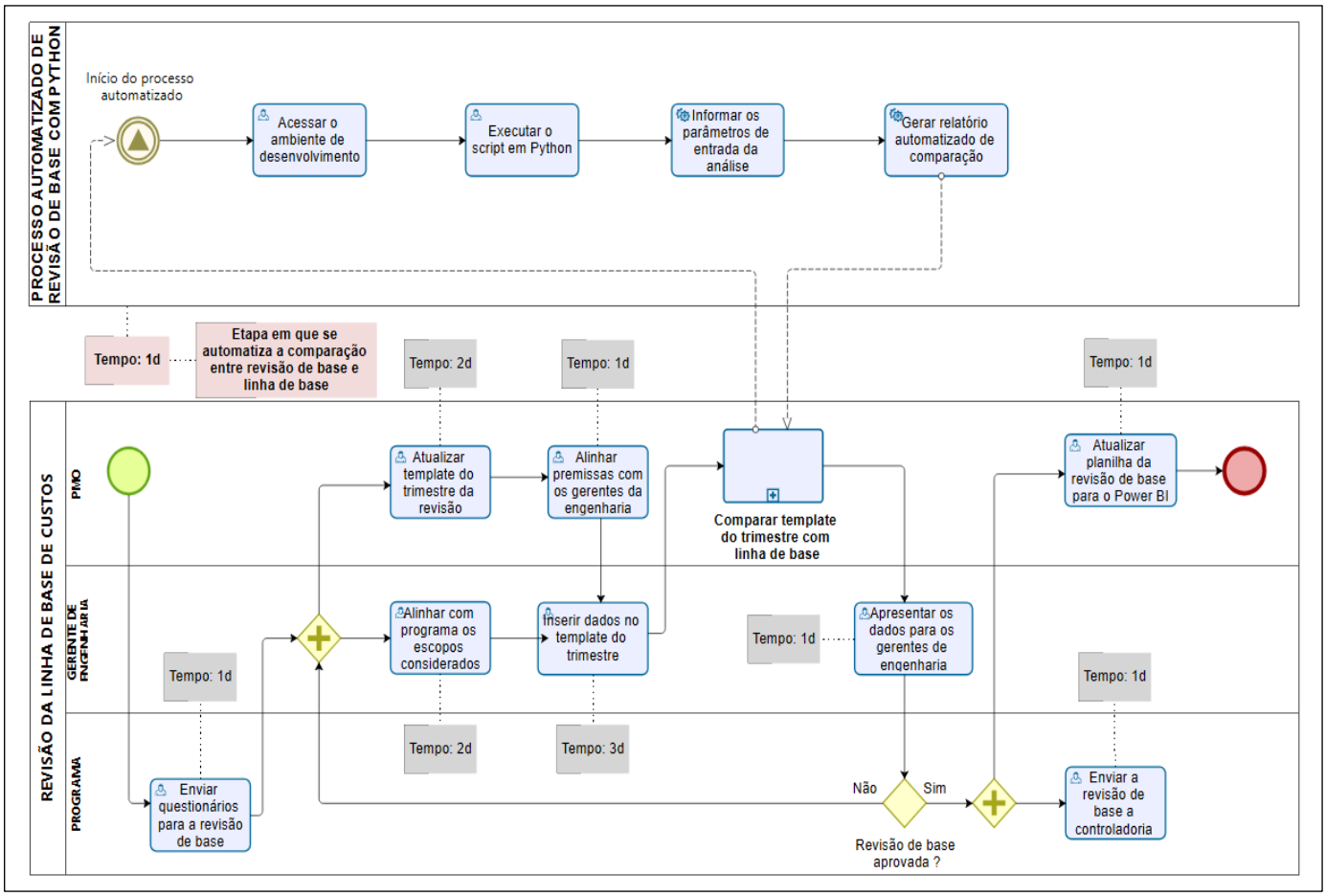
Dessa forma, a Figura 2 reforça visualmente a identificação das lacunas que justificam a melhoria proposta. A principal oportunidade está na automatização da comparação entre o *template* do trimestre e a linha de base, com o objetivo de reduzir esforço manual, tempo de execução e atividades corretivas posteriores.

3.3. Proposição do processo futuro (*TO-BE*)

Com base na análise do modelo *AS-IS* e nas lacunas encontradas, foi proposto um modelo futuro (*TO-BE*) para tornar o fluxo mais simples e reduzir o *lead time* no subprocesso de Revisão da Linha de Base de Custos. A principal mudança está na automatização da comparação entre o *template* do trimestre e a linha de base. Com essa melhoria, espera-se diminuir o esforço manual, reduzir validações e ajustes posteriores e tornar o subprocesso mais eficiente. A Figura 3 apresenta a modelagem do processo futuro em BPMN já com a reorganização do fluxo e a inclusão da automação em Python.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 3. Modelagem do subprocesso de Revisão da Linha de Base de Custos no estado futuro (*TO-BE*)



Fonte: Autoria própria, 2026.

A Figura 3 apresenta o subprocesso no estado futuro (*TO-BE*), evidenciando a reorganização do fluxo após a inclusão da automação em Python na etapa de comparação entre a revisão de base e a linha de base.

Conforme mostrado na Figura 3, o modelo *TO-BE* mantém as atividades essenciais do subprocesso e automatiza a comparação anteriormente executada de forma manual. Com isso, as etapas posteriores de validação e ajuste deixam de compor o fluxo proposto, reduzindo o *lead time* estimado.

A Tabela 3 apresenta, nesse novo modelo, o subprocesso com *lead time* estimado de 13 dias e as melhorias obtidas em cada atividade.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 3. Distribuição do *lead time* por atividade do subprocesso no estado futuro (*TO-BE*)

Atividade do Processo	Código da Atividade	*Lead time* Estimado (dias)
Enviar questionários para a revisão de base	AT01	1
Atualizar *template* do trimestre da revisão	AT02	2
Alinhar com o programa os escopos considerados	AT03	2
Alinhar premissas com os gerentes da engenharia	AT04	1
Inserir dados no *template* do trimestre	AT05	3
Comparar *template* do trimestre com linha de base	AT06	1
Validar análise com os gerentes de engenharia	AT07	0
Ajustar discrepâncias da análise	AT08	0
Apresentar os dados para os gerentes de engenharia	AT09	1
Enviar a revisão de base à controladoria de programa	AT10	1
Atualizar planilha da revisão de base para o Power BI	AT11	1
Ciclo total do processo *TO-BE*	**CT *TO-BE***	**13**

Fonte: Autoria própria, 2026.

3.4. Desenvolvimento da solução automatizada em Python

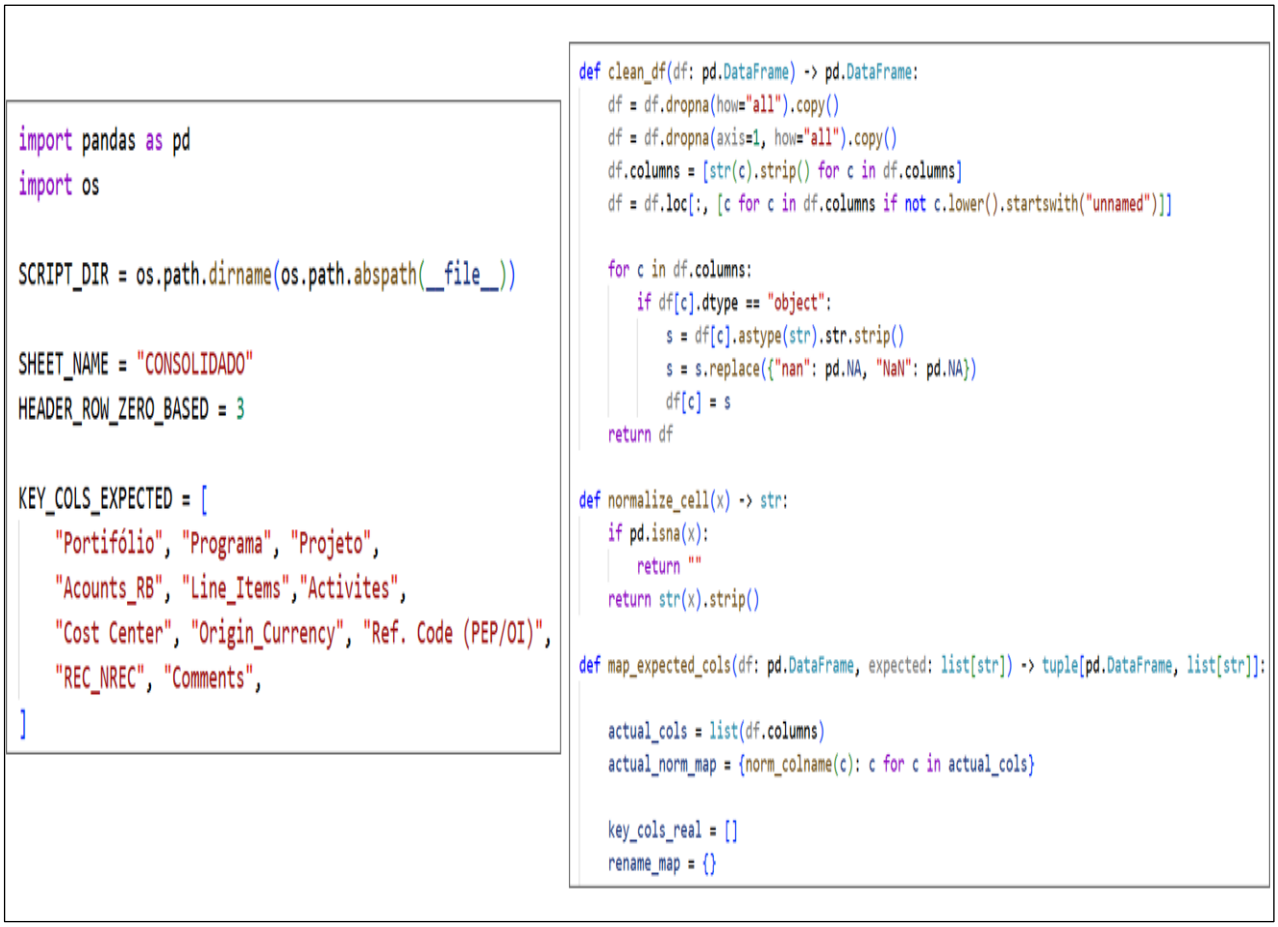
Após a definição do modelo *TO-BE*, foi desenvolvida uma solução em Python para automatizar a comparação entre a revisão de base e a linha de base. A implementação foi organizada em três etapas principais: preparação dos dados, comparação dos registros e geração do relatório automatizado.

3.4.1. Configuração da leitura, tratamento e padronização dos dados

Inicialmente, o *script* define os parâmetros de leitura das planilhas, como aba, linha de cabeçalho e colunas-chave para identificação dos registros. Na sequência, realiza o tratamento e a padronização dos dados, ajustando nomes de colunas, removendo estruturas vazias e organizando os valores. Essas operações reduzem inconsistências de formatação e tornam a comparação entre os arquivos mais uniforme e confiável. A Figura 4 apresenta os trechos de código correspondentes à configuração da leitura, ao tratamento e à padronização dos dados.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 4. Configuração da leitura, tratamento e padronização dos dados



```python
import pandas as pd
import os

SCRIPT_DIR = os.path.dirname(os.path.abspath(__file__))

SHEET_NAME = "CONSOLIDADO"
HEADER_ROW_ZERO_BASED = 3

KEY_COLS_EXPECTED = [
    "Portifólio", "Programa", "Projeto",
    "Acounts_RB", "Line_Items","Activites",
    "Cost Center", "Origin_Currency", "Ref. Code (PEP/OI)",
    "REC_NREC", "Comments",
]
```

```python
def clean_df(df: pd.DataFrame) -> pd.DataFrame:
    df = df.dropna(how="all").copy()
    df = df.dropna(axis=1, how="all").copy()
    df.columns = [str(c).strip() for c in df.columns]
    df = df.loc[:, [c for c in df.columns if not c.lower().startswith("unnamed")]]

    for c in df.columns:
        if df[c].dtype == "object":
            s = df[c].astype(str).str.strip()
            s = s.replace({"nan": pd.NA, "NaN": pd.NA})
            df[c] = s
    return df

def normalize_cell(x) -> str:
    if pd.isna(x):
        return ""
    return str(x).strip()

def map_expected_cols(df: pd.DataFrame, expected: list[str]) -> tuple[pd.DataFrame, list[str]]:

    actual_cols = list(df.columns)
    actual_norm_map = {norm_colname(c): c for c in actual_cols}

    key_cols_real = []
    rename_map = {}
```

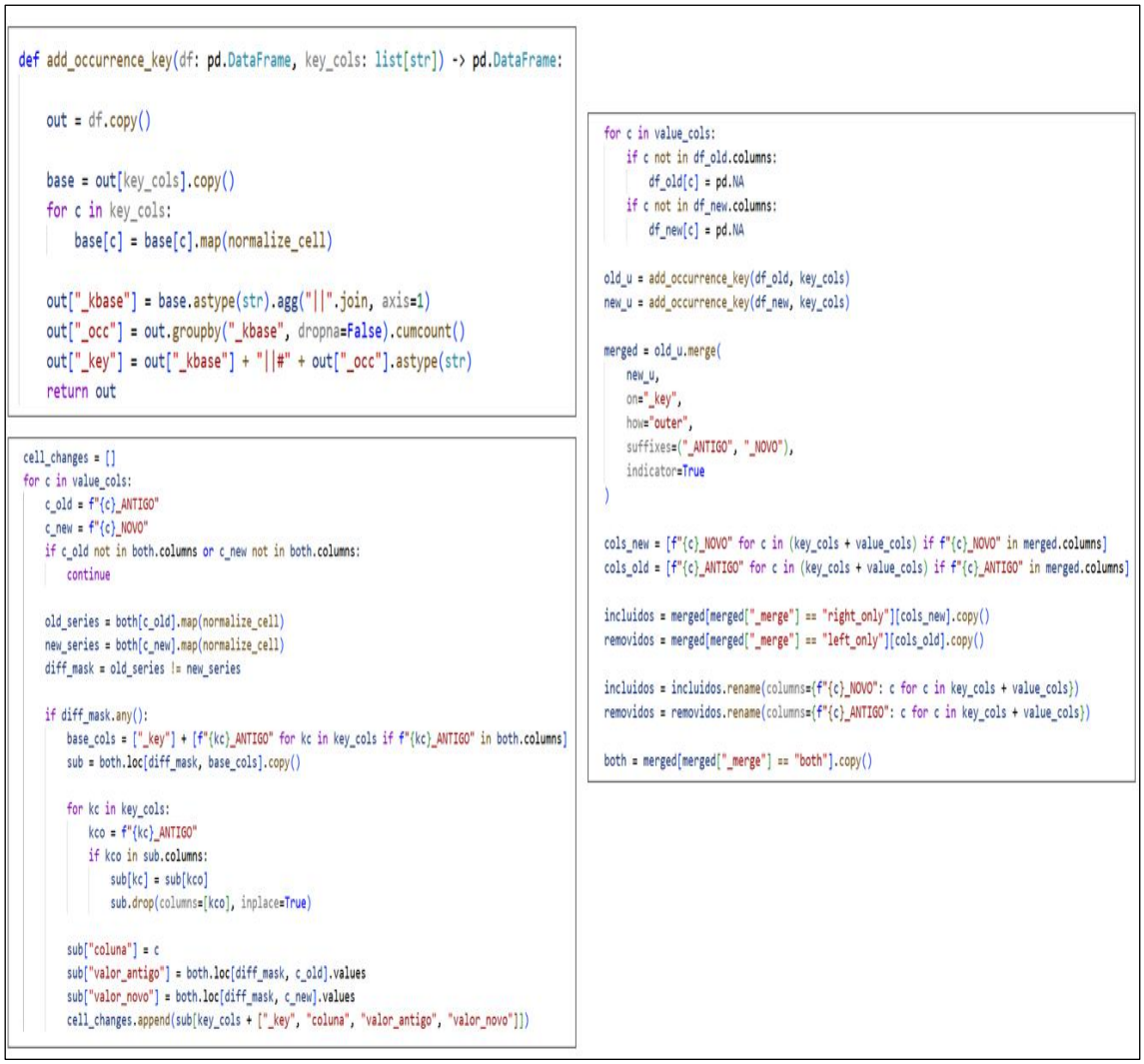
Fonte: Autoria própria, 2026.

3.4.2. Identificação dos registros e comparação entre linha de base e revisão

O *script* gera uma chave de identificação para cada registro, permitindo relacionar corretamente os registros da linha de base aos da revisão. Com essa referência, realiza a comparação entre os arquivos e identifica registros mantidos, incluídos, removidos ou alterados. As modificações também são verificadas em nível de célula, registrando os campos alterados e seus valores anteriores e posteriores. A Figura 5 apresenta os trechos de código correspondentes à identificação dos registros, à comparação entre linha de base e revisão e à identificação das alterações em nível de célula.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 5. Identificação e comparação dos registros entre linha de base e revisão



```
def add_occurrence_key(df: pd.DataFrame, key_cols: list[str]) -> pd.DataFrame:

    out = df.copy()

    base = out[key_cols].copy()
    for c in key_cols:
        base[c] = base[c].map(normalize_cell)

    out["_kbase"] = base.astype(str).agg("||".join, axis=1)
    out["_occ"] = out.groupby("_kbase", dropna=False).cumcount()
    out["_key"] = out["_kbase"] + "||#" + out["_occ"].astype(str)
    return out
```

```
for c in value_cols:
    if c not in df_old.columns:
        df_old[c] = pd.NA
    if c not in df_new.columns:
        df_new[c] = pd.NA

old_u = add_occurrence_key(df_old, key_cols)
new_u = add_occurrence_key(df_new, key_cols)

merged = old_u.merge(
    new_u,
    on="_key",
    how="outer",
    suffixes=("_ANTIGO", "_NOVO"),
    indicator=True
)

cols_new = [f"{c}_NOVO" for c in (key_cols + value_cols) if f"{c}_NOVO" in merged.columns]
cols_old = [f"{c}_ANTIGO" for c in (key_cols + value_cols) if f"{c}_ANTIGO" in merged.columns]

incluidos = merged[merged["_merge"] == "right_only"][cols_new].copy()
removidos = merged[merged["_merge"] == "left_only"][cols_old].copy()

incluidos = incluidos.rename(columns={f"{c}_NOVO": c for c in key_cols + value_cols})
removidos = removidos.rename(columns={f"{c}_ANTIGO": c for c in key_cols + value_cols})

both = merged[merged["_merge"] == "both"].copy()
```

```
cell_changes = []
for c in value_cols:
    c_old = f"{c}_ANTIGO"
    c_new = f"{c}_NOVO"
    if c_old not in both.columns or c_new not in both.columns:
        continue

    old_series = both[c_old].map(normalize_cell)
    new_series = both[c_new].map(normalize_cell)
    diff_mask = old_series != new_series

    if diff_mask.any():
        base_cols = ["_key"] + [f"{kc}_ANTIGO" for kc in key_cols if f"{kc}_ANTIGO" in both.columns]
        sub = both.loc[diff_mask, base_cols].copy()

        for kc in key_cols:
            kco = f"{kc}_ANTIGO"
            if kco in sub.columns:
                sub[kc] = sub[kco]
                sub.drop(columns=[kco], inplace=True)

        sub["coluna"] = c
        sub["valor_antigo"] = both.loc[diff_mask, c_old].values
        sub["valor_novo"] = both.loc[diff_mask, c_new].values
        cell_changes.append(sub[key_cols + ["_key", "coluna", "valor_antigo", "valor_novo"]])
```

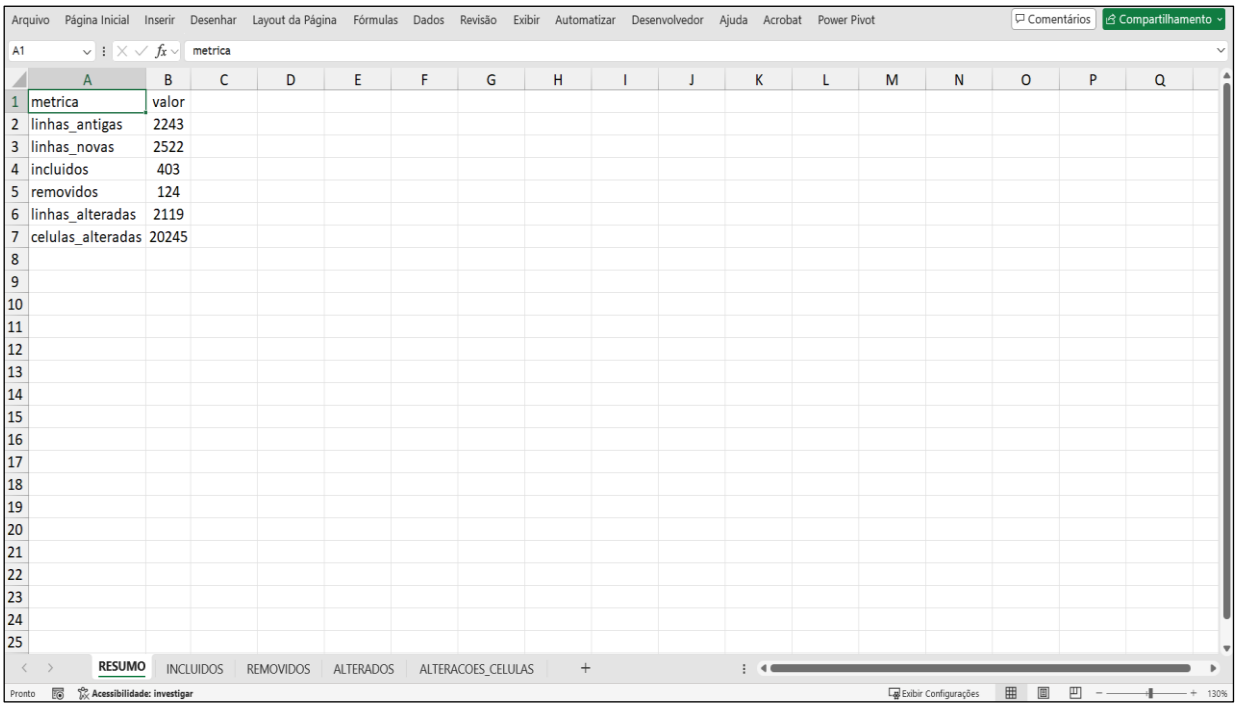
Fonte: Autoria própria, 2026.

3.4.3. Geração do relatório automatizado

Os resultados são consolidados automaticamente em um arquivo Excel com abas de resumo, registros incluídos, removidos, alterados e alterações em nível de célula. Essa saída organiza as evidências da comparação em formato estruturado e facilita a conferência e a validação do subprocesso. A Figura 6 apresenta um exemplo do relatório gerado.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 6. Relatório gerado automaticamente pelo *script* de comparação



Fonte: Autoria própria, 2026.

A solução desenvolvida em Python concretiza a mudança proposta no modelo *TO-BE* ao transformar a comparação entre a revisão de base e a linha de base em um procedimento padronizado e reproduzível, com saída estruturada para apoiar a execução do subprocesso.

3.5. Análise dos resultados

A comparação entre os modelos *AS-IS* e *TO-BE* demonstrou que a implementação da automação proposta resultou em uma redução significativa no tempo de execução do subprocesso.

O *lead time* total passou de 25 para 13 dias, ou seja, houve uma diminuição de 12 dias no ciclo. As atividades de validação da análise e ajuste de discrepâncias, que antes somavam 8 dias, foram removidas do fluxo. A etapa de comparação entre o *template* do trimestre e a linha de base também teve uma queda expressiva, passando de 5 dias para 1 dia. Com esses resultados, observa-se um processo mais simples, com menos correções e menor esforço operacional.

Com base nessa comparação, a Tabela 4 apresenta a síntese dos principais indicadores utilizados na avaliação da melhoria proposta.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 4. Síntese comparativa dos indicadores de desempenho do subprocesso

Indicador	Modelo *AS-IS*	Modelo *TO-BE*	Resultado
Lead time total do subprocesso	25 dias	13 dias	48% de redução
Tempo associado a retrabalho	8 dias	0 dias	100% de redução
Participação do retrabalho no processo	32%	0%	redução de 32 pontos percentuais
Tempo da atividade de comparação	5 dias	1 dia	80% de redução

Fonte: Autoria própria, 2026.

Os dados da Tabela 4 mostram que a redução de 12 dias no *lead time* total corresponde a 48%. As atividades de validação da análise e ajuste de discrepâncias, que somavam 8 dias no modelo *AS-IS*, foram eliminadas no *TO-BE*; por isso, o tempo associado ao retrabalho passou de 8 para 0 dias, correspondendo a uma redução de 100%. No cenário original, essas duas atividades representavam 32% do *lead time* total do subprocesso.

A atividade central de comparação (AT06) também apresentou ganho expressivo, passando de 5 para 1 dia, equivalente a uma redução de 80%. O resultado indica que a melhoria está diretamente associada à automatização da etapa que concentrava maior esforço manual e, em conjunto com a retirada das atividades corretivas posteriores, explica a redução observada no ciclo total.

De forma geral, os resultados confirmam que a proposta não se limita à implementação de um *script* isolado, mas representa uma melhoria do subprocesso apoiada pela modelagem BPMN e pela reorganização do fluxo entre os estados *AS-IS* e *TO-BE*.

4. CONSIDERAÇÕES FINAIS

Durante o estudo, observou-se que as etapas ligadas à comparação manual entre a revisão de base e a linha de base concentravam parte relevante do tempo do subprocesso. A modelagem *AS-IS* permitiu localizar esses pontos críticos e orientar a construção do modelo *TO-BE* com automação da atividade central de comparação.

A comparação entre os dois modelos evidenciou melhora no desempenho do subprocesso: o *lead time* estimado foi reduzido de 25 para 13 dias, o tempo associado ao retrabalho passou de 8 para 0 dias e a atividade central de comparação foi reduzida de 5 para 1 dia. Esses resultados indicam um fluxo mais simples, com menor dependência de conferências e correções posteriores.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Os resultados também reforçam que a automação, quando integrada à gestão de processos, pode produzir efeitos mais amplos do que a simples execução automática de uma tarefa. Nesse caso, a BPMN apoiou a compreensão e o redesenho do fluxo, enquanto Python viabilizou a padronização da comparação e a geração de informações estruturadas.

Como limitação, destaca-se que a solução foi desenvolvida para um contexto específico e depende da estrutura das planilhas analisadas. Sua aplicação em outros cenários pode exigir ajustes nas regras de comparação, nos campos de identificação e nas fontes de dados. Trabalhos futuros podem avaliar a integração da solução com outras bases e mecanismos de monitoramento contínuo do processo.

REFERÊNCIAS

AGANETTE, Karina de Jesus Pinto. **Business Process Management: estudo dos gatilhos organizacionais que motivam a implementação de sua prática e metodologia**. 2021. 166 f. Dissertação (Mestrado em Gestão & Organização do Conhecimento) – Escola de Ciência da Informação, Universidade Federal de Minas Gerais, Belo Horizonte, 2021. Disponível em: https://repositorio.ufmg.br/server/api/core/bitstreams/ea514e02-60d6-4473-8751-7c4f2b8983cf/content. Acesso em: 5 fev. 2026.

AMARILLA, Rosemara Santos Deniz; IAROZINSKI NETO, Alfredo. Análise comparativa dos principais processos de negócio de empresas do subsetor de edificações da construção civil. *Gestão & Produção*, v. 25, n. 2, p. 269-283, 2018. DOI: 10.1590/0104-530X2406-16. Disponível em: https://www.scielo.br/j/gp/a/s6WhPwwkCtDGrNkgtNDGDmb/?lang=pt. Acesso em: 5 fev. 2026.

ARAÚJO, Wagner Junqueira de; GOMES, Tamara Aureliano. Avaliação de sistemas de gerenciamento de processos de negócios (BPMS): análise multicritério dos softwares Bizagi e Bonita. *RDBCI: Revista Digital de Biblioteconomia e Ciência da Informação*, Campinas, SP, v. 20, n. 00, e022023, 2022. DOI: 10.20396/rdbci.v20i00.8670814. Disponível em: https://www.scielo.br/j/rdbci/a/599m74SL5Gy5hfwhhs64q9v/?format=html&lang=pt. Acesso em: 16 fev. 2026.

BOGNI, Gabriel Permonian; AVESANI, Isabella Maria; FERREIRA, Esther Eleotério Panequi; NANZER, Mateus Henrique Platinetti; BRUSCHI, Sarita Mazzini; OLIVEIRA, Vilma A. Algoritmos para os indicadores de fluxo de alunos para avaliar a evasão dos cursos. In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA, 50., 2022. **Anais [...]**, 2022. p. 1-12. DOI: 10.37702/2175-957X.COBENGE.2022.3955. Disponível em: https://admin.abenge.org.br/dados/COBENGE22/COBENGE22_00298_00003955.pdf. Acesso em: 30 out. 2025.

CUNHA, Ramon de Souza. **Automatização de processos contábeis: a aplicação de Python para a reclassificação de contas: transformando tarefas manuais em processos automatizados para maior produtividade**. 2024. 35 f. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Sistemas de Computação) – Instituto de Computação, Universidade Federal Fluminense, Rio de Janeiro, 2024. Disponível em:

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

https://app.uff.br/riuff/bitstream/handle/1/37461/TCC_RAMON_DE_SOUZA_CUNHA.pdf?sequence=1&isAllowed=y. Acesso em: 21 fev. 2026.

DALLILO, Felipe Diniz. **Uma contribuição metodológica à realização de ensaios de proficiência em software**. 2025. 214 f. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2025. DOI: 10.11606/T.55.2025.tde-16012026-134040. Disponível em: https://www.teses.usp.br/teses/disponiveis/55/55134/tde-16012026-134040/publico/FelipeDinizDallilo_DO_revisada.pdf. Acesso em: 19 mar. 2026.

GUEDES, Lucas Guaragna. **Gerenciamento de processos de negócio com BPM e automatização de atividades e tarefas com Python**. 2022. 92 f. Trabalho de Conclusão de Curso (Graduação em Engenharia Biomédica) – Universidade Federal de Uberlândia, Uberlândia, 2022. Disponível em: https://repositorio.ufu.br/bitstream/123456789/35918/1/GerenciamentoProcessoNeg%c3%b3cios.pdf. Acesso em: 28 mar. 2026.

GUIMARÃES, Paula Márcia da Silva; BRAGA, José Luiz; ZAIDAN, Fernando Hadad. Modelo de gerenciamento de processos de negócio integrado ao gerenciamento de riscos para a melhoria do processo produtivo. **Exacta**, São Paulo, v. 19, n. 4, p. 890-910, 2021. DOI: 10.5585/exactaep.2021.16023. Disponível em: https://periodicos.uninove.br/exacta/article/view/16023/8922. Acesso em: 28 mar. 2026.

LIEVANO-MARTÍNEZ, Federico A.; FERNÁNDEZ-LEDESMA, Javier D.; BURGOS, Daniel; BRANCH-BEDOYA, John W.; JIMENEZ-BUILES, Jovani A. Intelligent process automation: an application in manufacturing industry. *Sustainability*, Basel, v. 14, n. 14, art. 8804, 2022. DOI: 10.3390/su14148804. Disponível em: https://www.mdpi.com/2071-1050/14/14/8804. Acesso em: 14 mar. 2026.

TIERNEY, Gregory. A Python-based automation script to mark computer-aided design assessments. *Applied Sciences*, Basel, v. 15, n. 3, art. 1203, 2025. DOI: 10.3390/app15031203. Disponível em: https://www.mdpi.com/2076-3417/15/3/1203. Acesso em: 14 mar. 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.