Julius: Ferramenta de Captura de Dados dos Portais da Transparência

Pedro P. Berger[1], Matheus W. Souza[1], Heitor Quartezani[1], Guilherme Merisio[1], Iara A. Fazolo[1], Luciana G. F. de Andrade[1], Sandro T. Silva[1]
[1]Ministério Público do Estado do Espírito Santo (MPES)
Espírito Santo, Brasil
{pberger, mwsouza, ifazolo, landrade, stsilva}@mpes.mp.br

Resumo. *A ferramenta captura e consolida dados de APIs de portais de transparência dos 78 municípios do estado do Espírito Santo, auxiliando em investigações sobre compras públicas e contratos. O projeto enfrentou desafios técnicos, como a diversidade das APIs e a necessidade de padronização dos dados. A aplicação está disponível em código aberto e entrega um banco de dados públicos com todo o conteúdo dos portais da transparência de forma estruturada.*

Abstract. *The tool captures and consolidates data from transparency portal APIs of the 78 municipalities in the state of Espírito Santo, assisting in investigations of public procurement and contracts. The project faced technical challenges, such as the diversity of APIs and the need for data standardization. The application is available as open-source and provides a public database containing all the content from the transparency portals in a structured format.*

1. Introdução

O Os portais da transparência são ferramentas essenciais e obrigatórias para todas as instituições públicas, conforme estabelecido pela Lei 12.527/2011 [Brasil 2011]. Esses portais trazem inúmeros benefícios ao permitir a fiscalização, o controle social e a transparência dos gastos públicos. Os itens a seguir são comumente encontrados nos portais da transparência, e cada um deles representa uma seção específica com informações sobre determinado assunto relacionado às atividades de uma entidade governamental, quais sejam: Licitações e Contratos, Atas (documentos que registram os principais fatos e decisões de uma reunião ou sessão pública), Ordem de compras, Materiais entrada/saída, Bens consolidados (bens patrimoniais da entidade governamental, tanto móveis quanto imóveis), Frota de veículos, Receitas e Despesas (orçamento e execução), Empenhos (compromisso de gastos assumido pelo órgão público), Transferências Intraorçamentárias e Servidores.

Diversos programas foram propostos para fomentar a implantação da Lei de Acesso à Informação, tanto por órgãos de controle, como o Ministério Público, a Controladoria-Geral da União (CGU) e os Tribunais de Contas, quanto por organizações não governamentais. Esses esforços resultaram em um aumento significativo da transparência e no aprimoramento dos portais [Platt Neto 2023].

Atualmente, é comum encontrar portais com ferramentas dinâmicas de análise de informações e disponibilização dos dados para consumo automatizado por ferramentas computacionais, como *webservices* e APIs (*Application Programming Interface*). Nesse

contexto, o estado do Espírito Santo destaca-se ao alcançar a totalidade dos seus municípios com APIs para a disponibilização de dados públicos.

Inicialmente desenvolvido para identificar servidores públicos com acúmulo de cargos públicos, o Julius é a ferramenta utilizada pelo Ministério Público do Estado do Espírito Santo para capturar dados das APIs de todos os portais de transparência e consolidá-los, de forma automatizada, em um banco de dados unificado. Essa integração e automação proporcionam maior eficiência no acesso e na análise das informações públicas, facilitando a fiscalização e promovendo uma cultura de transparência. Este relato de experiência tem como objetivo apresentar a implementação e os benefícios do Julius, demonstrando como a tecnologia pode ser um aliado poderoso na promoção da transparência pública.

2. Arquitetura e Funcionamento

A ferramenta Julius consome as APIs de todos os portais de transparência implementados nos 78 municípios do estado do Espírito Santo, armazenando esses dados em um banco de dados SQLite [SQLite 2024]. Esse banco é posteriormente exportado e tratado, permitindo a integração com as ferramentas institucionais de defesa do patrimônio público e combate à corrupção.

A arquitetura da solução é composta por várias camadas e componentes. O estado do Espírito Santo conta com cinco diferentes empresas que desenvolvem os portais de transparência para os municípios, resultando em cinco diferentes APIs. Para gerenciar essa diversidade, a ferramenta Julius foi desenvolvida com a capacidade de interagir com cada uma dessas APIs, independentemente de suas especificidades.

A solução é desenvolvida em Python [Python 2024] e R [R Core team 2024] e executada em um container Docker [Docker inc. 2024], que faz o download e a execução dos códigos necessários para a captura dos dados, seguindo a estrutura apresentada na figura 1. Isso garante a portabilidade e a facilidade de implantação em diferentes ambientes. A ferramenta foi totalmente desenvolvida em código aberto, e o código-fonte pode ser acessado e está documentado no repositório do GitHub: https://github.com/mpes-uis/julius [MPES 2024].

O funcionamento da solução envolve várias etapas:

1. Consumo das APIs: A ferramenta Julius realiza chamadas periódicas às APIs dos portais de transparência, coletando dados atualizados sobre licitações, contratos, atas, ordens de compra, materiais, bens consolidados, bens móveis, frota, orçamento, receitas, despesas, empenhos, liquidações, pagamentos, transferências orçamentárias e servidores.
2. Armazenamento dos Dados: Os dados coletados são armazenados em um banco de dados SQLite, que atualmente possui um volume de 44 GB de dados.
3. Tratamento e Integração dos Dados: Após a coleta, os dados são exportados e tratados para garantir a integridade e a consistência. Esse processo inclui a normalização e a integração dos dados, permitindo a utilização pelas ferramentas institucionais.

4. Execução Automatizada: Toda a operação é automatizada por meio de scripts que são executados no ambiente Docker, garantindo que a coleta e o processamento dos dados ocorram de maneira contínua e eficiente.

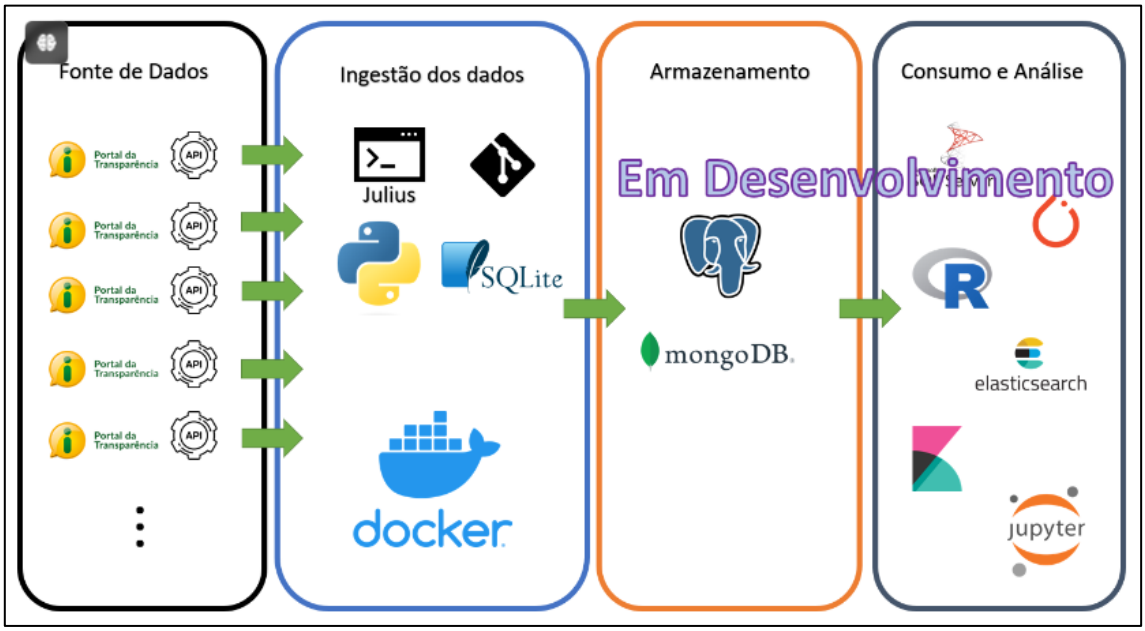

Figura 1. *Workflow* de funcionamento do Julius, fonte: https://github.com/mpes-uis/julius

3. Modelo de dados

Os dados coletados das APIs dos portais são tratados e armazenados em um banco de dados SQLite. A escolha pelo SQLite se deve à sua eficiência em aplicações de Análise de Dados, permitindo consultas SQL em arquivos de dados sem a necessidade de um servidor. Além disso, o SQLite possibilita análises mais complexas utilizando scripts simples escritos em linguagens como Python (que já inclui o SQLite embutido), R, ou outras linguagens que possuem adaptadores disponíveis.

O código foi desenvolvido com a intenção de ser facilmente replicável, considerando que a fonte de dados são APIs de portais de transparência disponíveis em grande parte dos órgãos públicos brasileiros. Dessa forma, o SQLite facilita a criação das estruturas de dados de acordo com o formato dos dados recebidos, tornando-as adaptáveis às informações fornecidas pelas APIs. Em outras palavras, a estrutura das tabelas é definida pelos atributos disponibilizados por cada API.

4. Desafios de desenvolvimento

O desenvolvimento do Julius enfrentou uma série de desafios técnicos complexos, que demandaram soluções inovadoras para garantir a eficiência e a eficácia da ferramenta. Um dos principais obstáculos foi a diversidade de portais da transparência, cada um desenvolvido por diferentes empresas e operando com APIs distintas. Essa heterogeneidade exigiu a criação de códigos específicos para cada API, o que aumentou significativamente a complexidade do projeto. Além disso, foi necessário desenvolver

mecanismos que permitissem a padronização dos dados coletados, gerando tabelas uniformes independentemente da API de origem.

Outro desafio crucial foi a implementação de sistemas de verificação da integridade dos dados consumidos. Frequentemente, as páginas dos portais da transparência estavam "offline" durante a execução da varredura automatizada, resultando em dados incompletos. Para contornar essa questão, o Julius foi equipado com funcionalidades que sinalizam automaticamente esses casos, permitindo que os usuários identifiquem e tratem os dados faltantes de forma eficiente.

Por fim, a necessidade de tornar o código modular e facilmente adaptável foi uma prioridade desde o início do desenvolvimento. Isso não apenas facilita a manutenção e a atualização da ferramenta, mas também permite que o Julius seja ajustado rapidamente para integrar novas APIs ou lidar com mudanças nos portais existentes, assegurando sua longevidade e relevância na promoção da transparência pública.

5. Sumario dos dados

A ferramenta Julius opera em uma base mensal, realizando a extração e o armazenamento dos dados de cada portal da transparência de forma contínua, mês a mês. Este processo segue o padrão estabelecido pelas APIs dos portais, permitindo a extração de diversas entidades, incluindo: leituras, licitações, contratos, atas, ordens de compra, entradas e saídas de materiais, bens consolidados, bens móveis, bens imóveis, frota de veículos, orçamento de receitas, execução de receitas, orçamento de despesas, empenhos, liquidações, pagamentos, transferências extraorçamentárias, transferências intraorçamentárias, servidores, plano de carreira, diárias e cargos de confiança.

É importante ressaltar que as entidades extraídas podem variar conforme o município, uma vez que cada município pode optar por alimentar ou não determinadas informações no portal, mesmo que a API correspondente esteja disponível para consumo pela empresa desenvolvedora do portal. Além disso, ocorrem casos em que municípios possuem entidades exclusivas ou temporárias. Por exemplo, durante a pandemia da COVID-19, alguns municípios criaram seções separadas para contratos relacionados à crise sanitária. Recentemente, um município passou a disponibilizar uma API específica para notas fiscais e convênios.

Cada entidade extraída é representada por uma tabela que contém diferentes campos. A quantidade e a natureza desses campos variam de acordo com a empresa responsável pelo desenvolvimento do portal ou conforme o interesse do município em disponibilizar determinadas informações. Um exemplo das tabelas e dos campos gravados em um banco de dados SQLite está apresentado na Figura 2:

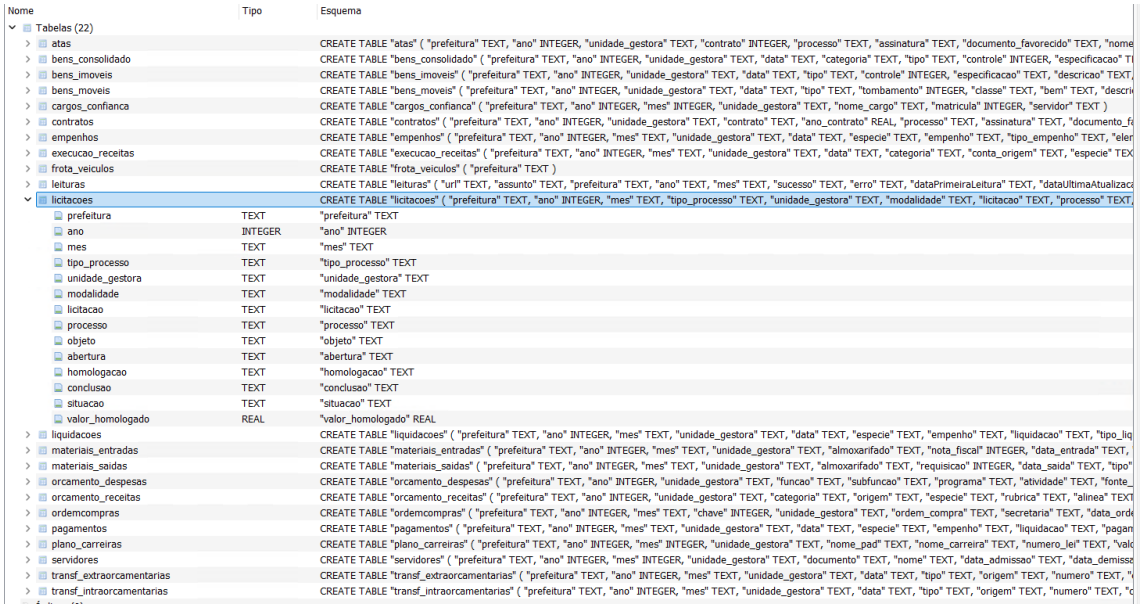

Figura 2: Apresentação dos dados armazenados em banco SQLite

5. Uso da ferramenta

Atualmente, a ferramenta Julius é utilizada pelo Ministério Público do Estado do Espírito Santo como um recurso fundamental para suporte em investigações, levantamento de informações, e está em fase de integração com os sistemas institucionais.

O uso inicial da ferramenta concentrou-se na identificação de servidores públicos que acumulavam cargos em diferentes municípios. Com a expansão da base de dados e a inclusão de novas tabelas, o Julius passou a ser empregado também na verificação autônoma de compras públicas e no rastreamento ágil de empresas e objetos envolvidos em processos de licitação. Além disso, a ferramenta tem sido progressivamente implementada em sistemas de investigação voltados para o combate à improbidade administrativa e à corrupção, destacando-se pela sua capacidade de identificar empresas com contratos públicos, facilitando a detecção de irregularidades.

6. Implementações futuras

Os próximos passos para o aprimoramento da ferramenta incluem a estruturação de banco de dados unificado já no processo de extração, convergindo os campos similares das diferentes APIs.

Também estudamos a captura e armazenamento de dados não estruturados, como editais e contratos. Para isso, está sendo planejada a utilização de soluções de bancos de dados NoSQL, que são mais adequados para lidar com a variabilidade e o volume desses dados. Além disso, estão sendo exploradas implementações de soluções de Processamento de Linguagem Natural (PLN) para facilitar o tratamento e a análise de textos, extraindo informações relevantes de documentos não estruturados.

Esses avanços visam ampliar a capacidade da ferramenta Julius, tornando-a ainda mais robusta e eficiente na promoção da transparência e no combate à corrupção no estado do Espírito Santo.

Referências

Brasil (2011). Lei nº 12.527, de 18 de novembro de 2011. Regula o acesso a informações previsto na Constituição Federal e dá outras providências. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm. Acesso em: 22 jul. 2024.

Docker Inc. (2024). Docker Documentation. Retrieved from https://docs.docker.com/

Merkel, D. (2014). Docker: Lightweight Linux Containers for Consistent Development and Deployment. Linux Journal, 2014(239), 2.

Ministério Público do Estado do Espírito Santo (2024). Julius: ferramenta de captura de dados. Disponível em: https://github.com/mpes-uis/julius. Acesso em: 22 jul. 2024.

PLATT NETO, O. A. (2023) Abordagem sistemática para avaliação de portais de transparência (ASAPoT). **Governet – Boletim de Orçamento e Finanças**, Curitiba, v. 19, n. 217, p. 412-417.

Python Software Foundation. (2024). Python Documentation. Retrieved from https://www.python.org/doc/

R Core Team. (2024). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Retrieved from https://www.r-project.org/

SQLite. (2024). SQLite Home Page. Retrieved from https://www.sqlite.org