Associação Propagadora Esdeva
Centro Universitário Academia – UniAcademia
Curso de Sistemas de Informação
Trabalho de Conclusão de Curso

Técnicas de Anonimização de Dados em Bancos NoSQL MongoDB

Matheus Reis Silva Soeiro Cabral[1]
Centro Universitário Academia, Juiz de Fora, MG

Tassio Ferenzini Martins Sirqueira[2]
Centro Universitário Academia, Juiz de Fora, MG

Linha de Pesquisa: Banco de Dados

RESUMO

[Contexto] Com o crescente volume de dados armazenados e processados por empresas e organizações, a proteção da privacidade dos dados tornou-se uma preocupação crucial. Bancos de dados NoSQL, conhecidos por sua capacidade de lidar com grandes volumes de dados não estruturados e semi-estruturados, são amplamente utilizados em aplicações que demandam alta escalabilidade e desempenho. Contudo, a natureza flexível e distribuída desses bancos de dados apresenta desafios significativos para a implementação de técnicas eficazes de anonimização, necessárias para proteger informações sensíveis contra acessos não autorizados. **[Objetivo]** O objetivo deste trabalho é investigar, avaliar e comparar diferentes técnicas de anonimização de dados aplicáveis a bancos de dados SQL (SQL Server e Mongo DB), visando identificar métodos que proporcionem um equilíbrio adequado entre segurança da informação e manutenção da usabilidade dos dados anonimizados. **[Metodologia]** A metodologia empregada neste estudo consiste em uma revisão das principais técnicas de anonimização de dados utilizadas em bancos de dados SQL e NoSQL, seguida de uma análise comparativa. Além disso, realizou-se a implementação de estudos de caso para avaliar a eficácia dessas técnicas em cenários reais, utilizando os bancos de dados SQL Server e MongoDB. **[Resultados]** Os resultados indicam que técnicas tradicionais de anonimização, como k-anonimato e l-diversidade, podem ser adaptadas para bancos de dados NoSQL com algumas modificações. Alternativamente, métodos específicos para NoSQL, como tokenização e criptografia também podem ser aplicados, embora apresentem desafios relacionados à complexidade de implementação e à gestão de chaves de criptografia. **[Considerações]** Este estudo aponta que, apesar das dificuldades inerentes à anonimização de dados em bancos, é possível alcançar um nível satisfatório de proteção da privacidade dos dados. No entanto, a escolha da técnica de anonimização deve ser cuidadosamente considerada, levando em conta o contexto específico de uso, os requisitos de segurança e as implicações de desempenho.

Palavras-chave: Anonimização de dados. Bancos NoSQL. MongoDB. Privacidade de

[1] Discente do Curso de Engenharia de Software do Centro Universitário Academia – UniAcademia. E-mail: matheusrsscabral2012@hotmail.com.
[2] Docente do Curso de Engenharia de Software do Centro Universitário Academia. Orientador.

dados.

ABSTRACT

[Context] With the growing volume of data stored and processed by companies and organizations, protecting data privacy has become a crucial concern. NoSQL databases, known for their ability to handle large volumes of unstructured and semi-structured data, are widely used in applications that demand high scalability and performance. However, the flexible and distributed nature of these databases presents significant challenges for implementing effective anonymization techniques necessary to protect sensitive information from unauthorized access. **[Objective]** The objective of this work is to investigate, evaluate and compare different data anonymization techniques applicable to SQL databases (SQL Server and Mongo DB), aiming to identify methods that provide an adequate balance between information security and maintaining the usability of anonymized data. **[Methodology]** The methodology used in this study consists of a review of the main data anonymization techniques used in SQL and NoSQL databases, followed by a comparative analysis of them. In addition, case studies were implemented to evaluate the effectiveness of these techniques in real scenarios, using the SQL Server and MongoDB databases. **[Results]** The results indicate that traditional anonymization techniques, such as k-anonymity and l-diversity, can be adapted to NoSQL databases with some modifications. Alternatively, NoSQL-specific methods such as tokenization and encryption can also be applied, although they present challenges related to implementation complexity and encryption key management. **[Considerations]** This study points out that, despite the difficulties inherent in anonymizing data in banks, it is possible to achieve a satisfactory level of data privacy protection. However, the choice of anonymization technique must be carefully considered, taking into account the specific context of use, security requirements and performance implications.

Keywords: Data anonymization. NoSQL databases. MongoDb. Data privacy.

1 INTRODUÇÃO

O avanço tecnológico e a transformação digital têm impulsionado o aumento exponencial do volume de dados gerados e armazenados por empresas e organizações.[3] Nesse cenário, os bancos de dados NoSQL surgiram como uma solução eficaz para gerenciar grandes quantidades de dados não estruturados e semi-estruturados, devido à sua flexibilidade, escalabilidade e alta performance.[4] Entretanto, com o aumento da coleta e armazenamento de dados, a proteção da privacidade das informações sensíveis tornou-se uma preocupação principal. A anonimização de dados,

[3] Disponível em: <https://fia.com.br/blog/transformacao-digital/>. Acesso em: 25/05/2024

[4] Disponível em: <https://www.researchgate.net/publication/327035187>. Acesso em: 18/04/2024.

que visa proteger a identidade dos indivíduos em conjuntos de dados, é essencial para assegurar a privacidade e conformidade com regulamentações de proteção de dados, como a LGPD (Lei Geral de Proteção de Dados).

Apesar da sua importância, a anonimização de dados em bancos NoSQL apresenta uma série de desafios. As técnicas de anonimização, desenvolvidas inicialmente para bancos de dados relacionais, muitas vezes não são diretamente aplicáveis ou eficientes em ambientes NoSQL. Há uma necessidade de adaptar e desenvolver novas técnicas de anonimização que sejam eficazes e eficientes para este tipo de banco de dados.

Este trabalho tem como objetivo principal investigar e avaliar as diferentes técnicas de anonimização de dados aplicáveis a bancos de dados NoSQL, identificando métodos que equilibrem a proteção da privacidade e a usabilidade dos dados.

Os objetivos específicos deste estudo são:

- Realizar uma revisão das principais técnicas de anonimização de dados.
- Analisar a aplicabilidade das técnicas de anonimização em bancos de dados SQL e NoSQL, mais especificamente (MongoDB).
- Desenvolver e testar adaptações dessas técnicas para SQL Server e Mongo Db.
- Comparar o uso das técnicas adaptadas.
- Apresentar um conjunto de códigos para realizar uma anonimização no MongoDb

A metodologia adotada neste estudo inclui uma das técnicas de anonimizaçãono ambiente SQL, seguida por uma técnica em ambiente NoSQL.

Este trabalho está dividido em cinco seções principais. A seção 2 apresenta a revisão das principais técnicas de anonimização e suas aplicações em bancos de dados. A seção 3 descreve a metodologia utilizada para a análise e testes das técnicas de anonimização no banco MongoDB. Na seção 4 são apresentados e discutidos os resultados obtidos nos estudos de caso. Por fim, na seção 5 aborda-se as considerações sobre o trabalho, as ameaças à validade e os trabalhos futuros.

2 REFERENCIAL TEÓRICO

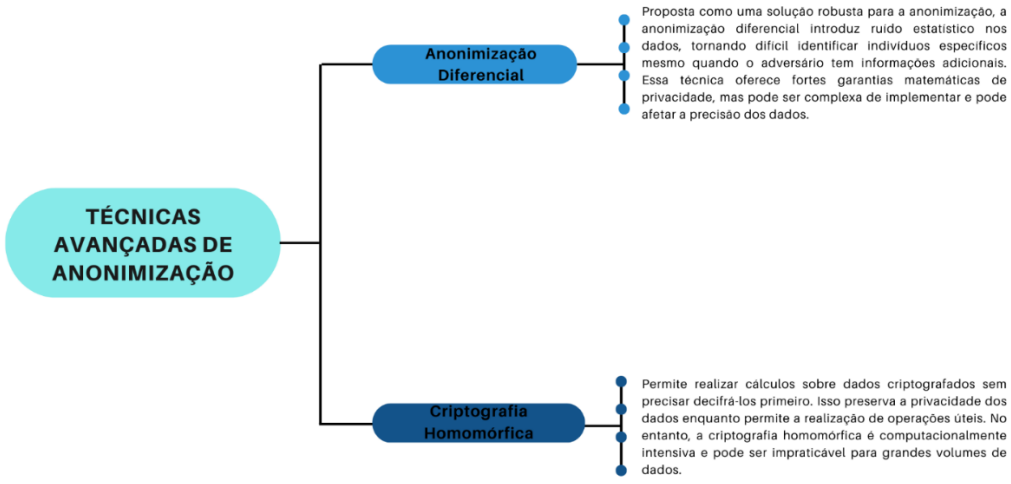
A anonimização de dados tem se tornado um campo de estudo muito discutido, especialmente com o aumento exponencial de dados gerados diariamente. Conforme destacado por Benjamin Fung (2010), existem várias técnicas de anonimização de dados, incluindo generalização, supressão, permutação e técnicas mais avançadas como a anonimização diferencial e a criptografia homomórfica. Cada técnica possui suas vantagens e limitações, e a escolha da abordagem adequada depende do contexto e dos requisitos específicos de privacidade e utilidade dos dados. A Figura 1 mostra a as técnicas tradicionais de anonimização e explica cada uma delas.

Figura 1. Técnicas Tradicionais de Anonimização.

Fonte: Elaborada pelo autor.

A Figura 2, são mostradas as técnicas avançadas de anonimização e explica-se cada uma delas.

Figura 2. Técnicas Avançadas de Anonimização.



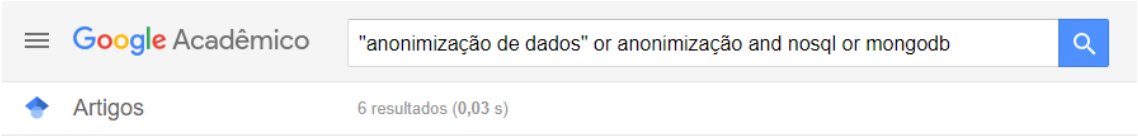
Fonte: Elaborada pelo autor.

3 METODOLOGIA

Na metodologia apresentada, foi realizado um levantamento de literatura e os critérios de inclusão e exclusão foram estabelecidos para garantir que apenas estudos relevantes e fossem considerados na revisão sistemática. Os critérios de exclusão que foram escolhidos são artigos que não discutem a anonimização de dados, apresentando a aplicação prática do MongoDB, um dos bancos NoSQL mais utilizados atualmente.

Tendo isso em vista, foram achados no total 6 resultados. Além disso, a preferência por artigos publicados em periódicos revisados por pares ou apresentados em conferências reconhecidas visa garantir a confiabilidade e validade dos resultados. Limitar a busca a publicações recentes também assegura a relevância e atualidade dos dados analisados, enquanto a exclusão de estudos duplicados ou não relevantes ajuda a manter a precisão e objetividade da revisão. Dos 6 resultados para embasar essa pesquisa, foram escolhidos três artigos.

Figura 3. String de Busca do Schollar Google.



Fonte: Elaborada pelo autor.

Começando pelo artigo 1, com o título de Anonymization of Clinical Data, este artigo aborda a anonimização de dados clínicos com o objetivo de proteger a privacidade dos pacientes enquanto permite a análise de dados para fins de pesquisa. O estudo foca no desenvolvimento de um método eficiente para anonimizar dados clínicos armazenados em MongoDB, utilizando modelos e algoritmos conhecidos de anonimização, como *k-anonymity*, *l-diversity* e *t-closeness*. A solução proposta foi validada usando dados clínicos reais e demonstrou um equilíbrio entre privacidade e valor de pesquisa.

No artigo 2, com o título de Proposta de Framework com Utilização de Big Data Baseado em Inteligência Competitiva para a Geração de Vantagem Competitiva, propõe-se um framework teórico que integra Big Data e inteligência competitiva para gerar vantagem competitiva em cadeias produtivas, com um foco particular na cadeia da carne bovina. O estudo detalha a coleta, processamento e distribuição de dados utilizando tecnologias de Big Data, incluindo o uso de MongoDB para armazenamento e análise.

Por fim, no artigo 3, com o Título Compreendendo o Conceito de Anonimização e Dado Anonimizado, explora-se o conceito de anonimização de dados, descrevendo várias técnicas como supressão, generalização, randomização e pseudoanonimização. O artigo discute os desafios da anonimização, incluindo a falibilidade das técnicas devido ao efeito mosaico, e a necessidade de uma avaliação contínua para garantir a eficácia

da anonimização.

Os três artigos escolhidos foram fundamentais para o desenvolvimento do TCC, devido à presença de aplicações práticas do MongoDB. Eles oferecem uma combinação de fundamentação teórica e estudos de caso práticos, demonstrando como diferentes técnicas de anonimização podem ser implementadas e validadas em ambientes de dados reais.

Os três artigos selecionados oferecem uma visão abrangente e detalhada sobre as técnicas de anonimização de dados em bancos NoSQL, com um foco particular na aplicação prática utilizando o MongoDB. Cada um dos estudos contribui de maneira única para entender como proteger a privacidade dos dados sensíveis.

A presença do MongoDB como plataforma de estudo em todos os artigos não apenas demonstra a versatilidade e adaptabilidade deste banco NoSQL para diferentes cenários de anonimização, mas também ilustra como as técnicas discutidas podem ser implementadas de forma eficaz para proteger a privacidade dos dados.

Em conjunto, esses estudos fornecem uma base sólida e atualizada para explorar e desenvolver soluções práticas e eficazes de anonimização de dados em ambientes NoSQL. A combinação de teoria e aplicação prática oferecida pelos artigos é essencial para enfrentar os desafios crescentes de privacidade de dados em um mundo digital cada vez mais interconectado e dependente de grandes volumes de informações. Foi feito também um estudo comparativo de técnicas de anonimização de dados em dois tipos distintos de bancos de dados: SQL Server e MongoDB.

Para o SQL Server, utilizou-se o recurso de *Dynamic Data Masking* (DDM), que permite mascarar dados sensíveis em tempo de consulta sem modificar os dados subjacentes. Os dados foram mascarados de tal forma que no campo do Email sejam mostrados os primeiros 2 caracteres, seguindo de anonimizar os dados até o “@”, e mantendo o resto do email visível. Já no campo de Telefone, serão anonimizados apenas os 4 primeiros dígitos do mesmo. Por fim no campo CPF, serão mostrados apenas o primeiro dígito e os dois últimos, sendo os do meio anonimizados.Foi demonstrado como aplicar máscaras nas colunas de uma tabela, como email, telefone e CPF, garantindo que os dados mascarados sejam exibidos apenas para usuários sem permissões privilegiadas.

No caso do MongoDB, a anonimização foi realizada através de um script Python utilizando a biblioteca pymongo[5]. Este script foi desenvolvido para percorrer documentos JSON em uma coleção MongoDB e aplicar transformações nos dados sensíveis, como email, telefone e CPF, antes de serem armazenados ou consultados. Foram aplicadas técnicas de Generalização e Supressão.

[5]<https://pymongo.readthedocs.io/en/stable/>. Acesso em 15/04/2024

Ambas as abordagens visam proteger a privacidade dos dados pessoais, impedindo a identificação direta dos indivíduos enquanto mantêm a utilidade dos dados para análise e pesquisa. A comparação entre SQL Server e MongoDB destacou as diferentes formas de implementar técnicas de anonimização, adaptadas às características específicas de cada banco de dados, como estrutura de dados, flexibilidade e escalabilidade.

Por isso aqui está um passo a passo para acompanhar melhor a comparação e a implementação da anonimização de dados nos dois bancos. Iniciando pela figura 4. Antes de iniciar a anonimização dos dados no SQL Server, é importante apresentar os dados em seu estado original. Então vamos considerar uma tabela chamada Clientes com as seguintes colunas: Id, Nome, Email, Telefone, CPF.

Figura 4. Apresentação do Dado Não-Anonimizado.

```
CREATE TABLE Clientes (
    Id INT PRIMARY KEY,
    Nome NVARCHAR(100),
    Email NVARCHAR(100),
    Telefone NVARCHAR(15),
    CPF NVARCHAR(11)
);

INSERT INTO Clientes (Id, Nome, Email, Telefone, CPF) VALUES
(1, 'João Silva', 'joao.silva@example.com', '1234567890', '12345678901'),
(2, 'Maria Oliveira', 'maria.oliveira@example.com', '0987654321', '98765432109');
```

Fonte: Elaborada pelo autor.

Na Figura 5, a anonimização será feita utilizando o recurso de Dynamic Data Masking (DDM) no SQL Server. Este recurso permite mascarar os dados sensíveis em tempo de consulta, sem modificar os dados subjacentes.

Figura 5. Aplicação da Anonimização com Dynamic Data Masking.

```
ALTER TABLE Clientes ALTER COLUMN Email ADD MASKED WITH (FUNCTION = 'partial(2,"*******@example.com",0)');

ALTER TABLE Clientes ALTER COLUMN Telefone ADD MASKED WITH (FUNCTION = 'partial(0,"****",4)');

ALTER TABLE Clientes ALTER COLUMN CPF ADD MASKED WITH (FUNCTION = 'partial(1,"****",2)');
```

Fonte: Elaborada pelo autor.

Na figura 6, realiza-se a consulta dos dados anonimizados. Para isto utilizamos os comandos SELECT e FROM, assim selecionamos quais dados queremos, e de qual tabela ele se origina.

Figura 6. Consulta aos Dados Anonimizados.

```
SELECT Id, Nome, Email, Telefone, CPF
FROM Clientes;
```

Fonte: Elaborada pelo autor.

Na Figura 7, os resultados exibidos para um usuário sem permissões privilegiadas serão mascarados. O Id, Nome e Email e CPF, porém podemos escolher livremente o que anonimizar, dependendo dos campos contidos na tabela.

Figura 7. Retorno SQL Server.

Resultados | Mensagens

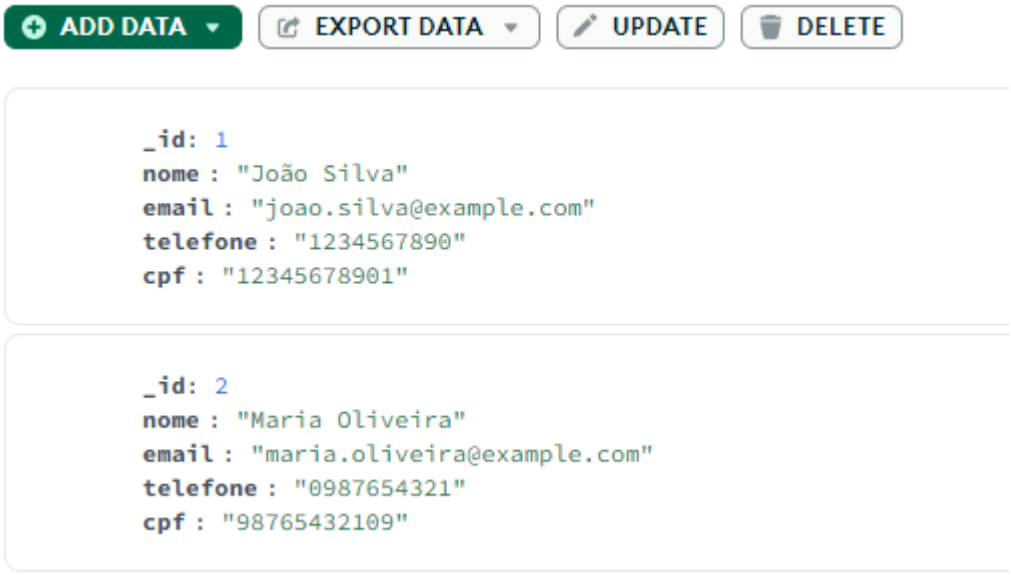
	Id	Nome	Email	Telefone	CPF
1	1	João Silva	jo********@example.com	****567890	1*******90
2	2	Maria Oliveira	ma********@example.com	****654321	9*******09

Fonte: Elaborada pelo autor.

Partindo para a parte do banco NoSQL, para realizar a anonimização dos dados no MongoDB, será feita a anonimização os campos Nome, Email, Telefone e CPF, seguindo o modelo de anonimização do SQL Server, porém com o diferencial que será um script feito em python, já que o SQL foi a inspiração do trabalho no MongoDb com os mesmos parâmetros.

No MongoDB, criou-se uma DataBase chamada “Clientes”, onde importou-se um arquivo JSON. Na figura 8, considere um documento JSON em uma coleção chamada clientes.

Figura 8. Importação do JSON para a Base de Dados.



Fonte: Elaborada pelo autor.

Agora, para anonimizar dados no MongoDB, uma abordagem comum é usar scripts que alteram os dados sensíveis. Vamos utilizar a biblioteca pymongo em Python para este propósito. Na figura 9, vemos Script Python para Anonimização, utilizando a Técnica DDM. Vale ressaltar que o script abaixo funciona apenas no Mongo Db, uma vez que ele é o objeto do estudo.

Figura 9. Script em Pyhton.

```python
from pymongo import MongoClient
import re

client = MongoClient('mongodb://localhost:27017/')
db = client['Clientes']
collection = db['clientes']

def mask_email(email):
    return re.sub(r'([^@]{2})[^@]*(@.*)', r'\1***\2', email)

def mask_phone(phone):
    return '****' + phone[4:]

def mask_cpf(cpf):
    return cpf[:1] + '*****' + cpf[-2:]

for cliente in collection.find():
    masked_email = mask_email(cliente['email'])
    masked_phone = mask_phone(cliente['telefone'])
    masked_cpf = mask_cpf(cliente['cpf'])

    print(f"Anonimização Realizada!")
    print(f"Email Anonimizado: {masked_email}")
    print(f"Telefone Anonimizado: {masked_phone}")
    print(f"CPF Anonimizado: {masked_cpf}")

    collection.update_one(
        {'_id': cliente['_id']},
        {'$set': {
            'email': masked_email,
            'telefone': masked_phone,
            'cpf': masked_cpf
        }}
    )
```

Fonte: Elaborada pelo autor.

Agora, na Figura 10, temos a verificação dos Dados Anonimizados e o retorno dos mesmos.

Figura 10. Dados Anonimizados.

```
PS C:\Users\First Place> & "C:/Users/First Place/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/First Place/Desktop/anonimizacao.py"
Anonimização Realizada!
Email Anonimizado: jo***@example.com
Telefone Anonimizado: ****567890
CPF Anonimizado: 1*****01
Anonimização Realizada!
Email Anonimizado: ma***@example.com
Telefone Anonimizado: ****654321
CPF Anonimizado: 9*****09
PS C:\Users\First Place>
```

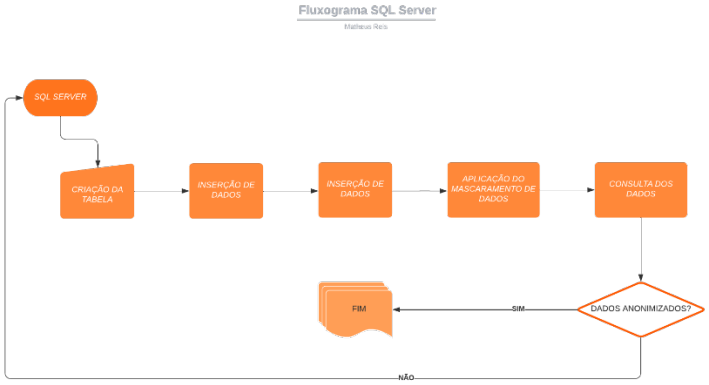
Fonte: Elaborada pelo autor.

4 RESULTADOS E DISCUSSÃO

A metodologia aplicada demonstrou a eficácia das técnicas de anonimização tanto no SQL Server quanto no MongoDB. No SQL Server, o uso de Dynamic Data Masking (DDM) facilitou a aplicação de máscaras nos dados sensíveis de forma eficiente e integrada. Essa abordagem é particularmente útil em cenários onde o controle de acesso aos dados é o principal foco, pois permite que diferentes níveis de usuários vejam dados mascarados ou completos conforme suas permissões.

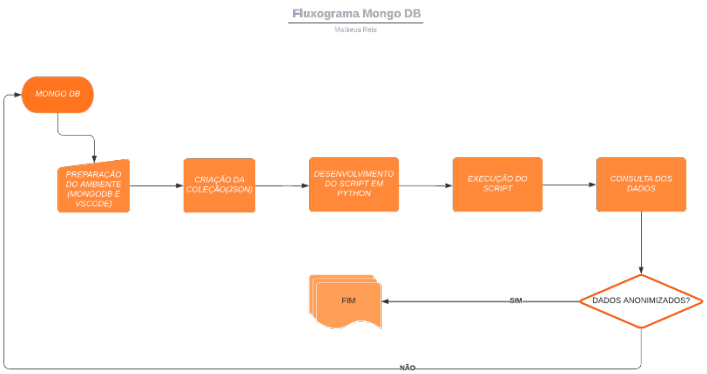
Por outro lado, a anonimização em MongoDB, realizada através de scripts personalizados em Python, assim como as técnicas de generalização e supressão, mostrou-se flexível e adaptável às necessidades específicas de anonimização. A capacidade de manipular documentos JSON e aplicar transformações diretamente nos dados permite uma personalização detalhada, essencial para lidar com diversos tipos de dados não estruturados. Nas figuras 11 e 12 pode-se observar o fluxograma de ambas as ferramentas a fins de comparação.

Figura 11. Fluxograma de uso da ferramenta SQL SERVER.



Fonte: Elaborada pelo Autor.

Figura 12. Fluxograma de uso da ferramenta MongoDB.



Fonte: Elaborada pelo Autor.

Ambas as técnicas se mostraram eficazes na proteção de dados sensíveis. No SQL Server, a DDM oferece uma solução pronta para uso que é fácil de implementar e gerenciar, garantindo que os dados mascarados sejam exibidos conforme necessário. No MongoDB, a abordagem baseada em scripts permite uma maior flexibilidade na aplicação de técnicas de anonimização, possibilitando a adaptação a diferentes requisitos de privacidade e utilidade dos dados.

A eficácia de cada técnica pode depender do contexto de uso. Para ambientes que requerem uma solução rápida e integrada com controle de acesso, o SQL Server com DDM se mostra eficaz. Já em ambientes que lidam com grandes volumes de dados não estruturados e necessitam de uma personalização avançada das técnicas de anonimização, o MongoDB se destaca.

Falando sobre os custos de implementação de técnicas de anonimização, ele pode variar entre SQL Server e MongoDB, como representado na figura 13.

Figura 13. Tabela de comparação de alocação de recursos entre SQL Server e MongoDB.

Característica	SQL Server	MongoDB
Custo de Implementação	**Baixo**	**Variável**
	A DDM (Dynamic Data Masking) é um recurso nativo que pode ser ativado sem a necessidade de ferramentas adicionais ou desenvolvimento extensivo.	Requer desenvolvimento de scripts personalizados, aumentando os custos iniciais devido à necessidade de programadores especializados.
Manutenção	**Simplificada**	**Complexa**
	A manutenção é simplificada pela integração nativa e suporte da Microsoft.	A manutenção pode ser mais complexa devido à necessidade de scripts personalizados e ao gerenciamento de um sistema flexível e escalável.
Flexibilidade e Escalabilidade	**Moderada**	**Alta**
	Embora ofereça boa escalabilidade, o SQL Server é mais rígido em comparação com o MongoDB e pode não ser tão flexível em manipular diferentes tipos de dados não estruturados.	O MongoDB oferece alta flexibilidade e capacidade de manipular grandes volumes de dados, o que pode compensar em projetos de longo prazo.

Fonte: Elaborada pelo Autor.
Disponível em:< https://www.microsoft.com/pt-br/sql-server/sql-server-2019-pricing>
<https://mongodb.com/pt-br/pricing>

O SQL Server tende a ser mais caro, especialmente para uso empresarial, mas oferece recursos robustos e integração com outros produtos da Microsoft. É uma escolha forte para empresas já investidas no ecossistema da Microsoft e que necessitam de capacidades abrangentes de banco de dados relacional.

Já o MongoDB, por outro lado, oferece mais flexibilidade com seus preços e é frequentemente mais barato para implantações em nuvem. É particularmente vantajoso para aplicações que requerem um banco de dados NoSQL e para aquelas que utilizam serviços em nuvem para escalabilidade.

Ambas as plataformas têm opções gratuitas adequadas para desenvolvimento e pequenos projetos, mas a escolha depende principalmente dos requisitos específicos do usuário ou da empresa em relação à estrutura do banco de dados, escala e orçamento.

A implantação de técnicas de anonimização em bancos de dados NoSQL como MongoDB pode trazer várias vantagens para o uso empresarial:

- Conformidade com Regulamentações: Empresas que lidam com dados sensíveis precisam estar em conformidade com a LGPD. A anonimização garante que os dados pessoais sejam protegidos contra acessos não autorizados.
- Análise Segura de Dados: Empresas podem realizar análises de dados em ambientes seguros, sem comprometer a privacidade dos indivíduos. Dados anonimizados permitem insights valiosos sem expor informações sensíveis.
- Flexibilidade de Dados: MongoDB permite que as empresas lidem com dados não estruturados de forma eficiente. A capacidade de aplicar técnicas de anonimização personalizadas ajuda a proteger a privacidade sem comprometer a utilidade dos dados.
- Escalabilidade: Empresas que lidam com grandes volumes de dados podem se beneficiar da escalabilidade do MongoDB, garantindo que as técnicas de anonimização sejam aplicadas eficientemente, mesmo em grande escala.

5 CONSIDERAÇÕES FINAIS

O presente trabalho analisou e implementou técnicas de anonimização de dados em dois tipos de bancos de dados: o relacional SQL Server e o NoSQL MongoDB.

Utilizando Dynamic Data Masking (DDM), o SQL Server provou ser uma solução robusta e integrada, permitindo a aplicação de máscaras de dados com facilidade e eficiência. A capacidade de gerenciar permissões detalhadamente assegura que diferentes níveis de usuários possam acessar os dados conforme suas necessidades, sem comprometer a segurança das informações. Esta técnica foi utilizada com base no site da Microsoft na parte de mascaramento de dados, então, dispondo disto, a fins de comparação, sabe-se que é uma técnica segura para mascaramento de dados e foi devidamente empregada.

Por outro lado, a anonimização de dados no MongoDB, realizada através de scripts Python, demonstrou grande flexibilidade, sendo capaz de lidar com dados não estruturados e exigências de anonimização personalizadas. A abordagem com MongoDB mostrou-se eficiente para grandes volumes de dados, destacando-se em cenários que demandam alta escalabilidade e desempenho. A implementação no SQL

Server tende a ter um custo inicial menor, devido às ferramentas nativas integradas. Já o MongoDB pode apresentar um custo maior de desenvolvimento inicial, mas oferece benefícios a longo prazo, especialmente em termos de flexibilidade e escalabilidade.

A anonimização de dados é essencial para empresas que lidam com grandes volumes de informações sensíveis. Suas aplicações incluem conformidade regulatória, ajudando as empresas a cumprirem com regulamentações de proteção de dados; segurança de dados, garantindo que informações sensíveis não sejam expostas indevidamente; análises e pesquisas, permitindo a realização de análises e pesquisas em dados de maneira segura e ética; e aumento da confiança dos clientes, ao assegurar que suas informações pessoais estão protegidas.

No decorrer do processo de desenvolvimento, algumas dificuldades foram encontradas, como a complexidade das técnicas de anonimização. A implementação de técnicas de anonimização eficazes, como generalização, supressão e perturbação, revelou-se desafiadora devido à necessidade de equilibrar a proteção da privacidade com a manutenção da utilidade dos dados. A escolha das técnicas adequadas para diferentes tipos de dados e cenários de uso exigiu um estudo mais aprofundado das características dos dados e dos objetivos analíticos.

Além disso, a integração com MongoDB apresentou dificuldades específicas, no quesito de scripts, uma vez que este banco de dados NoSQL possui uma estrutura flexível e dinâmica, diferente dos bancos de dados relacionais tradicionais. Adaptar as técnicas para funcionar de maneira eficaz em um ambiente NoSQL exigiu um esforço significativo em termos de pesquisa e desenvolvimento. O processamento de grandes volumes de dados para anonimização pode impactar a performance do sistema, e otimizar os processos de anonimização para minimizar o impacto no desempenho foi um desafio, embora isto seja impactado mais especialmente em cenários com grandes bases de dados.

Embora estas ferramentas possam ser de grande ajuda, é necessário compreender que elas possuem limitações e precisam ser melhoradas e evoluídas para alcançarem resultados cada vez mais relevantes. Muitas das ferramentas disponíveis para anonimização de dados são projetadas principalmente para bancos de dados relacionais e não suportam de maneira eficiente os bancos de dados NoSQL, limitando a aplicabilidade direta dessas ferramentas e exigindo adaptações significativas. Além disso, embora o MongoDB ofereça uma variedade de funcionalidades avançadas, algumas operações específicas de anonimização, como aquelas que requerem transformações complexas de dados, não são diretamente suportadas ou exigem abordagens menos eficientes. A escalabilidade das técnicas de anonimização em ambientes NoSQL de grande escala ainda é uma área em desenvolvimento, e ferramentas e métodos existentes podem não escalar de maneira eficiente para conjuntos de dados extremamente grandes, comuns em muitas aplicações de Big Data.

Como objetivos futuros para a ferramenta, pretende-se melhorar as seguintes funcionalidades: desenvolvimento de ferramentas avançadas de anonimização, criando

ferramentas específicas para MongoDb, com técnicas de machine learning para otimização; otimização de algoritmos de anonimização para melhorar a eficiência e reduzir tempo e recursos através de paralelização e computação distribuída; integração com tecnologias de Big Data; Adaptação de princípios de privacidade diferencial para bancos de dados NoSQL para maior segurança; desenvolvimento de APIs e bibliotecas que facilitem a implementação de técnicas de anonimização para desenvolvedores; melhoria na usabilidade e configuração, com interfaces mais intuitivas para configuração e uso de técnicas de anonimização; conformidade automática com regulamentações, garantindo que as ferramentas estejam em conformidade com regulamentações como a LGPD; combinação de anonimização com criptografia avançada, incluindo criptografia homomórfica; ferramentas que ofereçam feedback em tempo real sobre a eficácia das técnicas de anonimização; e colaboração interdisciplinar entre especialistas em segurança, desenvolvedores, cientistas de dados e legisladores para soluções mais eficazes.

Por fim, estas iniciativas visam desenvolver soluções mais eficientes, escaláveis e seguras para a anonimização de dados no MongoDb, atendendo às crescentes demandas de privacidade e análise de dados.

REFERÊNCIAS

ALTARADE, Mohammad. The Definitive Guide to NoSQL Databases. <https://www.toptal.com/database/the-definitive-guide-to-nosql-databases>. Acesso em: 04/07/2024

CASAROTTO, Eduardo Luís. **Anonimização de dados pessoais: uma abordagem baseada na lei e nas tecnologias de proteção de dados pessoais**. Disponível em: <https://repositorio.ufgd.edu.br/jspui/bitstream/prefix/2450/3/UFMS%20-%20EduardoLuisCasarotto.pdf>. Acesso em: 20/04/2024.

GHINITA, G.; KARRAS, P.; KALNIS, P.; MAMOULIS, N. Fast Data Anonymization with Low Information Loss. *Proceedings of the 33rd International Conference on Very Large Data Bases*, 2007. Disponível em: <https://cs.au.dk/~karras/vldb2007fawlil.pdf>. Acesso em: 18/04/2024.

LGPD. **Lei Geral de Proteção de Dados (LGPD)**. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>. Acesso em: 04/07/2024.

MELO, José Miguel. Privacidade e proteção de dados pessoais no contexto da inteligência artificial e big data. Disponível em: <https://repositorio-aberto.up.pt/bitstream/10216/106216/2/203764.pdf>. Acesso em: 20/04/2024.

MICROSOFT Documentation. Dynamic Data Masking. Disponível em: <https://learn.microsoft.com/pt-br/sql/relational-databases/security/dynamic-data-masking?view=sql-server-ver16>. Acesso em: 18/04/2024.

MONGODB Documentation. Data Masking. Disponível em: <https://www.mongodb.com/pt-br/docs/>. Acesso em: 18/04/2024.

Revista de Engenharia de Software Magazine 63. Disponível em: <https://www.devmedia.com.br/revista-engenharia-de-software-magazine-63/29391>. Acesso em: 18/04/2024.

SÃO PAULO (Estado). Escola Paulista da Magistratura. **Compreendendo o conceito de anonimização e dado anonimizado**. Disponível em: <https://www.tjsp.jus.br/download/EPM/Publicacoes/CadernosJuridicos/ii_9_anonimiza%C3%A7%C3%A3o_e_dado.pdf>. Acesso em: 20/04/2024.

SIRQUEIRA, Tassio; DALPRA, Humberto. NoSQL e a importância da Engenharia de Software e da Engenharia de Dados para o Big Data. 37º Jornada de Atualização da Informática (JAI). In: Congresso da Sociedade Brasileira de Computação (CSBC). Cap. 2018.

SWEENEY, L. k-Anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2002. Disponível em: <https://epic.org/wp-content/uploads/privacy/reidentification/Sweeney_Article.pdf>. Acesso em: 18/04/2024.