PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA
GRADUAÇÃO EM CIÊNCIA DE COMPUTAÇÃO

AUGUSTO LUIZ SANTOS QUEIROZ

BIG DATA

Consulta e cruzamento de dados de fontes estruturadas distintas

GOIÂNIA
2021

AUGUSTO LUIZ SANTOS QUEIROZ

BIG DATA

Consulta e cruzamento de dados de fontes estruturadas distintas

Trabalho de Conclusão de Curso apresentado à Escola Politécnica, da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Ciência da Computação.

Orientador:

Prof. Me. Max Gontijo de Oliveira

GOIÂNIA

2021

AUGUSTO LUIZ SANTOS QUEIROZ

BIG DATA

Consulta e cruzamento de dados de fontes estruturadas distintas

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de Bacharel em Ciência da Computação, e aprovado em sua forma final pela Escola Politécnica, da Pontifícia Universidade Católica de Goiás, em ____/____/_____.

_____

Profa. Ma. Ludmilla Reis Pinheiro dos Santos

Coordenadora de Trabalho de Conclusão de Curso

Banca examinadora:

_____

Orientador: Prof. Me. Max Gontijo de Oliveira

_____

Prof. Me. Fernando Gonçalves Abadia

_____

Profa. Dra. Carmen Cecília Centeno

GOIÂNIA

2021

Dedico este trabalho ao meu Deus e à minha família que sempre me apoiou em meus estudos.

AGRADECIMENTOS

A Deus, que está sempre colocando novos desafios na minha vida e me abençoando para que eu possa superá-los.

À minha esposa, Jhenyfer, por seu incentivo para que eu termine minha graduação, e sua compreensão, naqueles dias em que ficava sozinha, enquanto eu estava debruçado sobre o computador, estudando.

Aos meus pais, Agostinho e Maria do Carmo, que independentemente da situação financeira, insistiam para eu continuar estudando.

Ao meu irmão, Adriano, por muitas vezes ajudar em meus estudos. Inclusive, sentar-se ao meu lado e tentar entender o conteúdo das disciplinas.

Às minhas cunhadas, Hingridy e Gabriela, pela ajuda nos estudos.

Ao meu sogro, David e minha sogra Erlini, que me ajudaram na realização de tarefas que não conseguia realizar, pois estava estudando.

Ao meu professor e orientador Max, que me orientou quanto ao desenvolvimento do meu projeto, me passou bastante conhecimento e puxou minha orelha quando necessário.

Aos meus professores de curso, que cada um de seu jeito souberam me instruir e me ajudaram a formar o profissional que sou hoje.

“Na natureza nada se cria, nada se perde, tudo se transforma”

(Antoine-Laurent de Lavoisier)

RESUMO

Este trabalho trata sobre a construção de um Processador de Consulta. Este, por sua vez, irá realizar consultas e cruzamentos de dados em fontes de dados estruturadas distintas, com base no conceito de variedade do *Big Data*. O Processador de Consulta irá facilitar a análise de informações necessárias e essenciais ao usuário, no que diz respeito ao seu negócio.

O Processador de Consulta lê uma linguagem e processa o que está descrito nela. Este processamento é a consulta e o cruzamento de fontes de dados distintas e a aplicação possui algoritmos que leem essas fontes de dados separadamente e emite um resultado padronizado.

Este trabalho relata sobre a definição, estrutura e dificuldades de Big Data, composta de cinco Vs: Velocidade, Volume, Variedade, Veracidade e Valor, além dos tipos, cruzamentos e as estruturas de dados, ferramentas de análise de dados e os detalhes do desenvolvimento e do funcionamento do Processador de Consulta.

Palavras-chave: Big Data, Tipos de Dados, Dados Estruturados, Dados Semiestruturados, Dados Não Estruturados, Cruzamento de Tipos de Dados, Banco de Dados, Processador de Consulta, Fonte de Dados, Linguagem de Consulta Multifontes, Comando de Consulta.

ABSTRACT

This work deals with the construction of a Query Processor. This, in turn, will perform queries and data crossings in different structured data sources, based on the Big Data concept of variety. The Query Processor will facilitate the analysis of necessary and essential information to the user, regarding their business.

The Query Processor reads a language and processes what is described in it. This processing is the query and crossing of different data sources and the application has algorithms that read these data sources separately and output a standardized result.

This work reports on the definition, structure and difficulties of Big Data, composed of five Vs: Velocity, Volume, Variety, Veracity and Value, in addition to types, intersections and data structures, data analysis tools and development details and the functioning of the Query Processor.

Keywords: Big Data, Data Types, Structured Data, Semi-structured Data, Unstructured Data, Data Type Crossing, Database, Query Processor, Data Source, Multisource Query Language, Query Command.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

BI - *Business Intelligence*; em português: Inteligência empresarial

CRM - *Customer Relationship Management*; em português: Gestão de Relacionamento com o Cliente.

CSV - *Comma-Separated Values*; em português: Valores Separados por Vírgula

DER - Diagrama de Entidade e Relacionamento.

DSS - *Decision Support System*; em português: Sistema de Suporte à Decisão.

DW - *Data Warehouse*; em português: Armazém de Dados.

EDW - *Enterprise Data Warehouse*; em português: Armazém de Dados.

EIS - Executive Information System; em português: Sistema de Informação Executivo.

ETL - *Extract*, *Transform*, *Load*; em português: Extrair, Transformar, Carregar.

ERP - *Enterprise Resource Planning*; em português: Sistema de Gestão Empresarial.

GIS - *Geographic Information System*; em português: Sistema de Informação Geográfica

IoT - *Internet of Things*; em português: Internet das Coisas

JSON - *JavaScript Object Notation*; em português: Notação de Objetos JavaScript

NoSQL - *Not only* SQL; em português: Não somente SQL)

OLPT - *Online Transaction Processing*.

SGBD - Sistema de gerenciamento de banco de dados.

SGBDC - Sistema de gerenciamento de banco de dados colunar.

SGBDR - Sistema de gerenciamento de banco de dados relacional.

SQL - *Structured Query Language*; em português: Linguagem de Consulta Estruturada

TI - Tecnologia da Informação.

XML - *eXtensible Markup Language*; em português: Linguagem de Marcação Extensível

SUMÁRIO

1. INTRODUÇÃO

Já estamos vivendo o fechamento e o encerramento de empresas que não se prepararam no sentido de buscar informações de sua concorrência, não colocando em prática melhorias e ações, por não possuírem informações gerais do mundo digital e não conseguirem tomar ações de manutenção e sobrevivência competitiva. O sucesso de dar atenção ao que se denomina *Big Data* torna-se cada dia mais importante no que diz respeito a investimentos para esta nova arquitetura de metodologia, análise e ferramentas. Hoje, claramente, temos uma palavra forte no cenário da informática, a qual mobiliza muitas pessoas que pesquisam e buscam entender o que é, como se utiliza e como pode ser utilizado o fenômeno *Big Data* (Machado, 2018). Tão importante quanto gerar informação é a capacidade de processamento de dados volumosos em alta velocidade. Isso se comprova pelo fato de que, nas últimas décadas, presenciamos o desenvolvimento de supercomputadores que atendam essa necessidade: quanto mais a tecnologia foi penetrando no meio social, mais informações as pessoas foram gerando e consumindo *Big Data* (Machado, 2018). Uma das definições de *Big Data* é estruturada como sendo composta de cinco Vs: Velocidade, Volume, Variedade, Veracidade e Valor. Um dos grandes desafios do *Big Data* é trabalhar com uma grande variedade de formatos e estruturas que deverão ser conciliadas para as análises necessárias. É preciso integrar diversas fontes de dados para poder incluir novos tipos e estruturas de dados às fontes de dados com as quais a organização já trabalha. Os dados são gerados nos mais diversos tipos de formatos de dados estruturados e de dados não estruturados (Machado, 2018). Dados estruturados são organizados em linhas e colunas em formato de tabela. Quando são encontrados em bancos de dados relacionais, apresentam campos bem definidos (Neto, 2021). Dados não estruturados são dados que não possuem uma estrutura lógica de composição e organização. Esses dados requerem um pré-processamento para análise (Neto, 2021). O processo de integração envolve profissionais específicos, assim como desenvolvimento de aplicações de ferramentas de software que unifiquem os formatos ou criem formas de integração entre estes (Machado, 2018).

Assim, o objetivo desse trabalho é a construção de um Processador de Consulta que, por sua vez, realizará consultas e cruzamentos de dados em fontes de dados estruturadas distintas, com base no conceito de Variedade do *Big Data*. Este processador facilitará a análise de informações necessárias e essenciais ao usuário, no que diz respeito a um negócio, com consulta e cruzamento de dados de fontes estruturadas distintas.

A metodologia adotada foi a exploratória, pois foi necessário buscar conhecimento tanto para aprofundar quanto para obter noção do assunto abordado. As fontes primárias de informação utilizadas foram: na área de *Big Data*, (Machado, 2018) e (Neto, 2021); na área de linguagem de programação, (Cocian, 2004); Na área de dados, (Fry, 2008) e (Bassett, 2019). As fontes secundárias foram as 14

documentações do DBeaver e do BigQuery, além de sites de documentação de empresas.

O Capítulo I apresenta as definições de *Big Data*, a organização interna dos dados que são utilizados e os desafios dessa nova tecnologia. No Capítulo 2 é proporcionada fundamentação teórica para que os objetivos deste trabalho sejam alcançados. É mostrado na parte de *Big Data*, seus conceitos, os seus cinco Vs, suas dificuldades, seus tipos de dados e o cruzamento entre seus tipos de dados e ferramentas de análise de dados. Na parte de Dados, é mostrado as estruturas que os compõem, alguns exemplos de arquivos e bancos de dados específicos. No Capítulo 3 são apresentados os detalhes do desenvolvimento e do funcionamento do Processador de Consulta construído, seu objetivo, a definição da linguagem que ele interpreta, os resultados esperados e os materiais e métodos utilizados.

2. REFERENCIAL TEÓRICO

Este capítulo proporciona fundamentação teórica para que os objetivos deste trabalho sejam alcançados. Na parte de *Big Data*, são apresentados seus conceitos, os seus cinco Vs, suas dificuldades, seus tipos de dados e o cruzamento entre seus tipos de dados e ferramentas de análise de dados. Na parte de Dados, são mostradas as estruturas que os compõem, alguns exemplos de arquivos e bancos de dados específicos.

2.1 *BIG DATA*

Big Data é o termo em Tecnologia da Informação (TI), utilizado para descrever grandes volumes e variações de dados, que precisam ser processados em uma alta velocidade para gerar informações para maior visibilidade e tomada de decisão (Machado, 2018). Esse termo surgiu para se referir às aplicações de computadores que utilizam grandes volumes de dados em diferentes formatos, que podem ser agrupados, lidos, convertidos, analisados com técnicas estatísticas, matemáticas e computacionais, gerando um novo tipo de conhecimento chamado "*Data Insight*", algo conclusivo e ainda nunca pensado sobre os dados originais (Neto, 2021). *Big Data* ganha mais relevância à medida que há um aumento sem precedentes no número de dados gerados a cada dia. As tecnologias de *Big Data* trazem formas inovadoras de processamento de dados. Essas tecnologias descrevem uma nova geração de arquiteturas, projetadas para extrair valor de uma imensa variedade de dados, que permitem e necessitam de alta velocidade de processamento. O objetivo é capturá-los e analisá-los, de forma a transformá-los em informações importantes e valiosas no âmbito de gestão de negócios (Machado, 2018). Essas informações (*Insights*), geradas a partir dos dados, podem resultar em um sólido apoio na decisão de mudanças na orientação de negócios, ou na criação de um novo produto. Quando um novo produto é criado a partir de *Insights* ele é chamado de *Data-Driven Product* (Produto Orientado a Dados). *Insights* podem revolucionar a organização e seus negócios. As organizações que se valem desta tecnologia são conhecidas como *Data-Driven Companies*, em português, Companhia Orientada a Dados (Neto, 2021).

2.1.1 Os cinco Vs

Uma das definições de *Big Data* é estruturada como sendo composta de cinco Vs: Volume, Velocidade, Variedade, Veracidade e Valor (Machado, 2018).

Figura 1 - Os cinco Vs do *Big Data*

Fonte: Os cinco Vs do *Big Data*, 2017.

O termo Volume dentro do *Big Data* se refere à grande quantidade de dados. Uma única pessoa pode gerar uma infinidade de dados no seu dia a dia, que podem ser muito valiosos para a obtenção de valor. Esses dados podem ser suas preferências musicais, os aplicativos acessados através de um smartphone, o restaurante em que almoçou e o que comeu, os assuntos que tratou em redes sociais.

Quando se considera essa vasta quantidade de dados e se aplica a todos os milhões de usuários no mundo, pode se ter uma ideia do volume gigantesco destes, que podem ser coletados, analisados, tratados e aproveitados (Machado, 2018).

O termo Velocidade dentro do *Big Data* se refere à grande agilidade com que os dados são produzidos. Os dados são criados e fluem em uma velocidade nunca antes vista e devem ser tratados em tempo hábil.

Hoje em dia, para alguns negócios, como análises de dados médicos, detecção de fraudes e liberações de pagamentos, um minuto pode ser muito tempo, pois são negócios onde as informações são sensíveis ao tempo. Negócios de características semelhantes como estes possuem um grande potencial para se

encaixar em modelos de *Big Data*, tendo em vista a grande quantidade de dados produzidos e que precisam ser processados rapidamente. Já a maior parte dos projetos de *data warehouse* e *Business Intelligence* têm uma latência de 1 dia, ou seja, são informações do dia anterior (Machado, 2018).

O termo Variedade dentro do *Big Data* se refere à grande diversidade de dados que podem ser úteis para a geração de valor. Os dados são gerados nos mais diversos tipos de formatos de dados estruturados e de dados não estruturados. Os mesmos podem ser extraídos de bancos de dados relacionais, de planilhas, de textos em páginas de Internet e Intranet, de postagens em Redes Sociais, de dispositivos de georreferenciamento, de fotos e vídeos transmitidos em tempo real por um drone conectado à internet, por exemplo.

Enfim, são inúmeras as possibilidades de se obter dados. Devido a essa natureza de diversas fontes e formas de se representar os dados, tem-se aqui um grande desafio dentro do *Big Data*: lidar com essa variedade de dados, simultaneamente (Machado, 2018).

O termo Veracidade dentro do *Big Data* se refere à necessidade que as pessoas têm de garantir que os dados coletados de alguma fonte de dados sejam autênticos e verdadeiros naquele momento da coleta e obtenção dos mesmos. Redes Sociais são fontes de dados de *Big Data* e nem tudo que é postado por seus usuários é verdadeiro e confiável. Da mesma forma, sistemas de informação também são fontes de *Big Data*, porém, podem possuir dados com erros (Machado, 2018).

O termo Valor dentro do *Big Data* é o ponto mais destacado em relação a aplicações de *Big Data*. Não faz sentido aplicar os conceitos de *Big Data* se não for possível extrair valor útil dos dados para análises de negócios de uma organização. Em projetos *Big Data*, a estratégia de obtenção de valor a partir das informações analisadas deve ser bem definida. A principal prioridade e orientação é ter objetivos claros e metas bem definidas. Um projeto que não atinge um objetivo de geração de algo de valor para o negócio está fadado ao fracasso. Existem dados bons e dados ruins. Dados bons geram bons resultados, dados ruins geram resultados ruins (Machado, 2018).

2.1.2 Dificuldades

A aplicação dos conceitos de *Big Data* no negócio de uma organização não é algo trivial. De fato, inúmeras são as dificuldades encontradas em se trabalhar com *Big Data*. Nesta seção serão apresentadas algumas delas:

a) Armazenamento de dados e qualidade

Com o crescimento em ritmo acelerado de organizações, a quantidade de dados produzidos cresce também. O armazenamento dessa enorme quantidade de dados é um desafio muito grande. Para suprir essa demanda de armazenamento, um *data warehouse* pode tentar combinar dados não estruturados de diversas fontes. Mas aí surge um problema bem real: neste momento ele encontra erros,

dados ausentes, dados inconsistentes, conflitos lógicos, dados duplicados. Tudo isso resulta em conflitos lógicos de qualidade de dados.

As tecnologias de *Big Data* vêm evoluindo para suprir essa necessidade de armazenamento dessa imensa quantidade de dados que vêm surgindo. É importante que organizações adotem essas novas tecnologias para, assim, contornar esse problema de armazenamento e gerar *Insights* mais confiáveis (Synnex Westcon-Comstor, 2021).

b) Entendimento insuficiente e aceitação de *Big Data*

As organizações podem perder muito tempo e recursos em ferramentas que não sabem utilizar e em processos que não sabem conduzir. Para piorar a situação, colaboradores que não entendem o valor do *Big Data* podem não querer trabalhar em processos novos ou na mudança de processos existentes para sua adoção. Isso pode ocorrer não só com *Big Data*, mas, também, na adoção de qualquer outra tecnologia nova, ou processo novo. Esses colaboradores poderão resistir e impedir o progresso organizacional. Portanto, sem um entendimento claro do motivo de haver uma iniciativa em adotar um projeto *Big Data* em uma organização, o mesmo está fadado ao fracasso. Para contornar isso, os departamentos de TI precisam organizar treinamentos e *workshops* para mostrar o valor do *Big Data* (Synnex Westcon-Comstor, 2021).

c) Incerteza do gerenciamento de dados

Uma frente importante no gerenciamento de *Big Data* é o uso de uma ampla gama de ferramentas e estruturas inovadoras de gerenciamento de dados. Uma delas são as estruturas NoSQL - *Not only* SQL; em português: Não somente SQL). Estruturas NoSQL se diferem dos sistemas de gerenciamento de banco de dados relacionais (SGBDR) pois são amplamente projetadas para atender às demandas de desempenho de aplicativos de *Big Data*. Uma dessas demandas é gerenciar uma grande quantidade de dados em um breve tempo de resposta (Synnex Westcon-Comstor, 2021). Porém, um ponto forte dos SGBDR é a consistência dos dados e a integridade de transações. Ponto que o NoSQL não possui, o mesmo apenas garante o último valor atualizado, isso se nenhuma atualização for realizada até o momento da consulta (Bruno Rafael, 2019).

d) Segurança de dados

Em organizações que trabalham com *Big Data*, os dados obtidos podem ser originados de uma grande variedade de fontes, das quais algumas não são confiáveis, seguras, ou não são compatíveis com padrões organizacionais. Portanto, estes dados apresentam possíveis problemas de segurança. Para resolver essa questão, é necessário introduzir práticas de segurança de dados para coleta, armazenamento e recuperação desses dados (Synnex Westcon-Comstor, 2021).

e) Sincronização de fontes de dados

A sincronização de fontes de dados é uma demanda complicada, pois, além da árdua tarefa de extrair dados de diferentes fontes a fim de importá-los para plataformas de *Big Data*, ainda pode ocorrer desses dados terem diferentes taxas e agendamentos e podem sair rapidamente da sincronização com o sistema origem (Synnex Westcon-Comstor, 2021).

2.1.3 Tipos de Dados

As aplicações tradicionais de computadores utilizam dados estruturados, tabelas com linhas, colunas e campos bem definidos para processamento. Os dados na era do *Big Data* originam-se de diversos locais, com diferentes tipos e formatos. Isso torna complexa sua manipulação (Neto, 2021).

Em uma aplicação que se caracteriza como *Big Data*, os dados que ela manipula podem vir de diversas fontes e terem diferentes formas de estruturação. Uma parte desses dados poderia vir, por exemplo, de um Banco de Dados relacional; uma outra parte poderia vir de um Satélite e estar no formato do Sistema de Informação Geográfica (GIS); outros dados poderiam vir de arquivos de documento como Word ou planilhas como Excel; outros dados poderiam vir de de textos de Redes Sociais; dados bastante relevantes poderiam vir de sensores de máquinas. De uma mistura de dados como esta, pode-se produzir *Insights*, que conduzirão para a solução de problemas, à resposta a uma pergunta e, talvez, à criação de um novo produto (Neto, 2021).

Dados Estruturados, Semiestruturados e Não Estruturados são os tipos de dados existentes.

Dados Estruturados são organizados em linhas e colunas em formato de tabela. Quando são encontrados em bancos de dados relacionais apresentam campos bem definidos. Esses bancos de dados são muito eficientes na recuperação e processamento dos dados armazenados ali. Planilhas eletrônicas e a linguagem de consulta estruturada (SQL), usada nos bancos de dados relacionais, são meios próprios para acesso e manipulação desses dados estruturados (Neto, 2021).

Dados Semiestruturados são dados com organização diferenciada. É necessária uma prévia análise dos dados para identificação da estrutura desses formatos. Normalmente, dados semiestruturados são provenientes da Web, nos formatos XML e JSON (Neto, 2021). No caso do formato XML, se for utilizado o XML Schema, então os dados estarão estruturados.

Dados Não Estruturados são dados que não possuem uma estrutura lógica de composição e organização. São dados de vídeo, áudio, e-mails, documentos de textos em geral, por exemplo: post e blogs e dados gerados por aplicativos de Redes Sociais, como mensagens do WhatsApp e Twitter. Esses dados requerem um pré-processamento para análise (Neto, 2021). Ainda que os formatos de texto rico, como doc, docx, odt, tenham estrutura interna para descrever o documento, o teor dos dados não é descrito de maneira estruturada.

Dentro desse tipo, existem os Dados em Movimento (D*ata in Motion*), que referem-se a dados de *stream*, em trânsito, se movendo através da rede, de um nó para outro, ou seja, de um lado para outro. O termo *live-streaming*, por exemplo, se refere às transmissões ao vivo feitas através da internet. O processamento e a análise destes dados são feitos em tempo real, à medida que vão sendo capturados. Esses dados são difíceis de processar, pois têm custo maior. Porém, devido a sua importância, e se usados corretamente, podem proporcionar à organização obter conhecimentos valiosos em tempo real e, assim, obter vantagens estratégicas ou tomar decisões de grande impacto em tempo hábil. Por esse motivo, esses dados são uma parte importante do *Big Data*.

Existem também os Dados em Repouso (D*ata in Rest*). Esses dados estão armazenados em um destino estável. Não estão em uso e nem viajando para outros destinos. Uma vez que estes dados atingem seu destino, recebem camadas de proteção adicionais, como criptografia, proteção por senha e estruturas de permissão de acesso para os usuários. São um exemplo desses dados, os dados armazenados em *data warehouse*. Esses dados são muito importantes, pois contam a história da organização e fornecem a base para sua operação e existência.

Além dos mencionados, existe dentro do contexto de dados não estruturados, os Dados Pequenos (S*mall Data*). É um termo utilizado para se referir a pequenas quantidades de dados, mas que representam quantidade suficiente para uma tomada de decisão. É um dado que não ganha no volume, mas na qualidade. É um dado que já está pronto e limpo, para ser usado em uma análise de negócios. São exemplos de S*mall Data*, dados provenientes de sistemas ERP *(Enterprise Resource Planning*; em português, Sistema de Gestão Empresarial) ou CRM (*Customer Relationship Management*; em português: Gestão de Relacionamento com o Cliente). Tanto *Big Data* quanto S*mall Data* são soluções complementares, ambas têm um imenso significado na operação das organizações e podem ser benéficas quando utilizadas para resolver problemas comuns.

2.1.4 Cruzamento de Tipos de Dados

Um dos grandes desafios do *Big Data* é trabalhar com uma grande variedade de formatos e estruturas que deverão ser conciliadas para as análises necessárias. É preciso integrar diversas fontes de dados para poder incluir novos tipos e estruturas de dados (sociais, sensores, vídeo) às fontes de dados com as quais a organização já trabalha (bancos de dados relacionais, bancos de dados de mainframes legados) e talvez conciliar ainda com bancos de dados NoSQL. Este processo de integração e filtragem envolve profissionais específicos, assim como o desenvolvimento de aplicações de ferramentas de software que unifiquem os formatos ou criem formas de integração entre estes (Machado, 2018).

As figuras 2, 3, 4, 5 e 6 mostram um cenário possível de se obter um cruzamento de tipos de dados.

Representando a primeira fonte de dados, a Figura 2 é o resultado de uma consulta a uma tabela de nome pessoa, criada utilizando o SGBD PostgreSQL.

Essa tabela foi criada para fins de teste, seu intuito é o cadastro de pessoas e possui três colunas, que são: identificador, nome e cpf. Estas, têm o objetivo de armazenar o identificador (gerado automaticamente pelo SGBD), o nome e o cpf de cada pessoa cadastrada.

Representando a segunda fonte de dados, a Figura 3 é um arquivo CSV, onde se têm dados de pessoas. Esse arquivo foi também criado para fins de teste e possui duas colunas, que são: nome e sexo. Estas, têm o objetvo de armazenar o nome e o sexo de cada pessoa cadastrada.

Representando a terceira fonte de dados, a Figura 4 é o resultado, no formato de tuplas, do cruzamento da primeira e da segunda fonte de dados. Esta, possui as colunas e os dados da primeira e da segunda fonte de dados.

Indo mais além, representando a quarta fonte de dados, a Figura 5 é um arquivo JSON, onde se têm dados de pessoas. Esse arquivo foi também criado para fins de teste e possui um conjunto de pares chave-valor, de chaves: cpf e cnh. Estas, possuem os valores de nome e o sexo, respectivamente, de cada pessoa cadastrada.

Representando a quinta e última fonte de dados, a Figura 6 é o resultado, no formato de tuplas, do cruzamento da terceira e da quarta fonte de dados. Esta, possui as colunas e os dados da terceira e da quarta fonte de dados.

O cruzamento, no caso mostrado, foi realizado, pois ambas fontes de dados possuem uma coluna que funciona como elo de ligação entre elas, ou seja, a coluna nome. O cruzamento de tipos de dados é realizado quando as fontes de dados a serem cruzadas possuem atributos que interligam essas fontes de alguma forma. Esses atributos não precisam, necessariamente, ter o mesmo nome, ou serem do mesmo tipo.

Figura 2 - Consulta à tabela pessoa

	IDENTIFICADOR [PK] integer	NOME character varying (50)	CPF numeric (11)
1	1	Felipe Sérgio Silva	80213688964
2	2	João Thomas Cardoso	23492374352
3	3	Martin Fábio Galvão	51438511493
4	4	Ricardo Manoel Dias	56807943229
5	5	Isabela Sophia Peixoto	43659031607
6	6	Luciana Larissa Almeida	90897441931
7	7	Mirella Lívia Gonçalves	35049790204
8	8	Valentina Fátima da Rosa	61316172368

Fonte: Próprio Autor.

Figura 3 - Arquivo CSV com dados de pessoas

	A	B
1	NOME	SEXO
2	Danilo Renan Cavalcanti	Masculino
3	Felipe Sérgio Silva	Masculino
4	João Thomas Cardoso	Masculino
5	Martin Fábio Galvão	Masculino
6	Pedro Henrique Castro	Masculino
7	Ricardo Manoel Dias	Masculino
8	Emilly Melissa Cardoso	Feminino
9	Isabela Sophia Peixoto	Feminino
10	Jéssica Emilly da Cruz	Feminino
11	Luciana Larissa Almeida	Feminino
12	Mirella Lívia Gonçalves	Feminino
13	Valentina Fátima da Rosa	Feminino

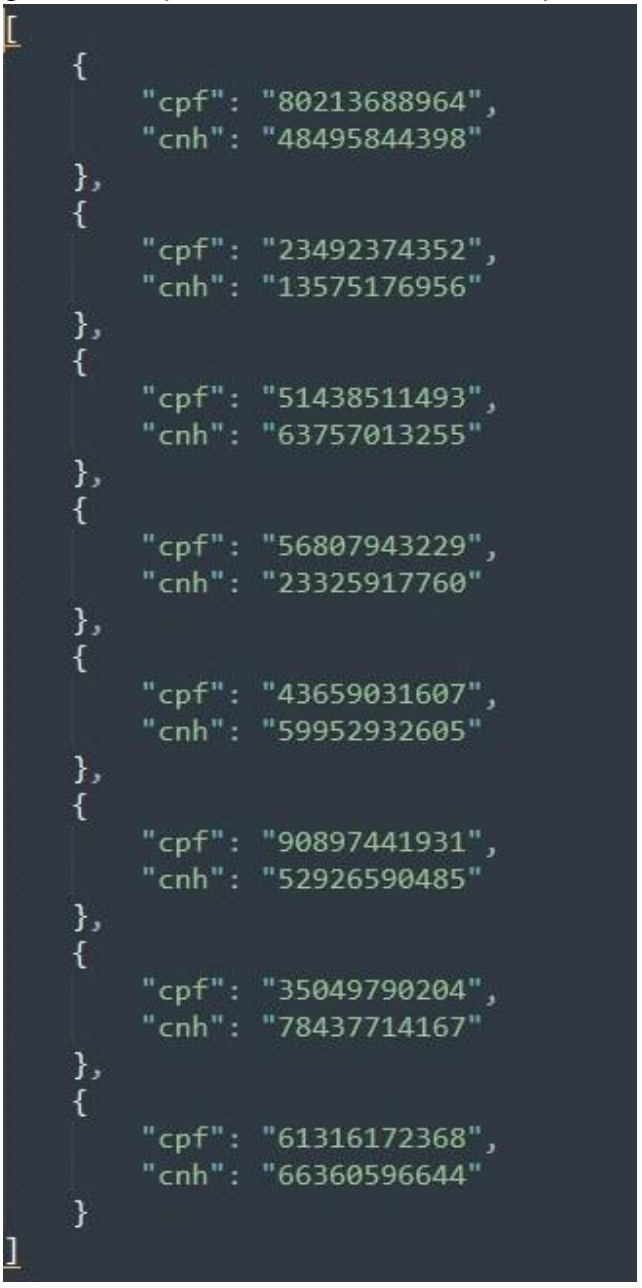
Fonte: Próprio Autor.

Figura 4 - Cruzamento da tabela pessoa com o CSV de pessoas

IDENTIFICADOR	NOME	CPF	NOME	SEXO
1	Felipe Sérgio Silva	80213688964	Felipe Sérgio Silva	Masculino
5	Isabela Sophia Peixoto	43659031607	Isabela Sophia Peixoto	Feminino
2	João Thomas Cardoso	23492374352	João Thomas Cardoso	Masculino
6	Luciana Larissa Almeida	90897441931	Luciana Larissa Almeida	Feminino
3	Martin Fábio Galvão	51438511493	Martin Fábio Galvão	Masculino
7	Mirella Lívia Gonçalves	35049790204	Mirella Lívia Gonçalves	Feminino
4	Ricardo Manoel Dias	56807943229	Ricardo Manoel Dias	Masculino
8	Valentina Fátima da Rosa	61316172368	Valentina Fátima da Rosa	Feminino

Fonte: Próprio Autor.

Figura 5 - Arquivo JSON com dados de pessoas



```
[
    {
        "cpf": "80213688964",
        "cnh": "48495844398"
    },
    {
        "cpf": "23492374352",
        "cnh": "13575176956"
    },
    {
        "cpf": "51438511493",
        "cnh": "63757013255"
    },
    {
        "cpf": "56807943229",
        "cnh": "23325917760"
    },
    {
        "cpf": "43659031607",
        "cnh": "59952932605"
    },
    {
        "cpf": "90897441931",
        "cnh": "52926590485"
    },
    {
        "cpf": "35049790204",
        "cnh": "78437714167"
    },
    {
        "cpf": "61316172368",
        "cnh": "66360596644"
    }
]
```


Fonte: Próprio Autor.

Figura 6 - Cruzamento da tabela pessoa com o CSV de pessoas e com o JSON de pessoas

IDENTIFICADOR	NOME	CPF	NOME	SEXO	CPF	CNH
1	Felipe Sérgio Silva	80213688964	Felipe Sérgio Silva	Masculino	80213688964	48495844398
5	Isabela Sophia Peixoto	43659031607	Isabela Sophia Peixoto	Feminino	43659031607	59952932605
2	João Thomas Cardoso	23492374352	João Thomas Cardoso	Masculino	23492374352	13575176956
6	Luciana Larissa Almeida	90897441931	Luciana Larissa Almeida	Feminino	90897441931	52926590485
3	Martin Fábio Galvão	51438511493	Martin Fábio Galvão	Masculino	51438511493	63757013255
7	Mirella Lívia Gonçalves	35049790204	Mirella Lívia Gonçalves	Feminino	35049790204	78437714167
4	Ricardo Manoel Dias	56807943229	Ricardo Manoel Dias	Masculino	56807943229	23325917760
8	Valentina Fátima da Rosa	61316172368	Valentina Fátima da Rosa	Feminino	61316172368	66360596644

Fonte: Próprio Autor.

2.1.5 *Big Data Analytics*

Big Data Analytics é um termo utilizado para definir um desdobramento do *Big Data*. Este desdobramento identifica não somente poderosos softwares capazes de tratarem os dados gerados com as tecnologias do *Big Data*, mas, também, as técnicas utilizadas, objetivando transformá-los em informações úteis às organizações.

Antes da criação, definição e utilização do processo *Big Data Analytics* para se analisar e gerar informações a partir de dados, eram utilizadas fórmulas matemáticas ou técnicas avançadas de probabilidades e estatística, por exemplo: análise de frequência, série histórica e estudos de médias móveis, as quais eram executadas manualmente e, obviamente em função das limitações humanas, trabalhando com um número reduzido de variáveis. A inovação tecnológica possibilitou a existência de processadores de alta capacidade e de altíssimas velocidades que proporcionam transitar todos esses cálculos, por meio de softwares desenvolvidos e orientados para transformação e processamento dos chamados “rastros digitais” em informações estratégicas de uma organização. Essas informações podem ser utilizadas em análises para obtenção de *insights* que podem levar, muito provavelmente, a organização a ter melhores decisões e direções estratégicas de negócio.

Utilizando-se do *Big Data Analytics* é possível analisar dados estruturados e não estruturados, isso vem facilitar a descoberta, em tempo real, de oportunidades e informações que estão muito além de relatórios rotineiros, às vezes, com dados do dia anterior, ou do fechamento do balanço do mês anterior, sobre dados existentes e produzidos por uma organização (Machado, 2018).

2.1.6 *Data Warehouse*

Data Warehouse (DW), ou *Enterprise Data Warehouse* (EDW), são armazéns de dados organizacionais consolidados, repositórios, tratados com níveis de segurança absolutos para garantir a integridade e a operação do negócio. DW permite a integração de dados corporativos distribuídos pelos nós da rede, capturando, armazenando dados e tornando-os acessíveis aos usuários de níveis decisórios.

Dados consolidados são dados corretos, reais, íntegros, que permitem análise e tomada de decisões baseadas em informações verdadeiras. Esses dados são capturados a partir de Sistemas Tradicionais, ou OLPT, que são sistemas de processamento online das operações da organização em vários níveis de negócio, tais como, entrada de pedidos, transações financeiras, relacionamento com clientes, vendas de varejo, entre outros. Centenas de usuários estão conectados a esses sistemas, gerando dados, realizando transações e negócios e obtendo resultados de suas buscas nas DW.

Organizações geram dados estruturados em grandes volumes, que precisam ser armazenados com segurança para uso diário. As DW são uma solução para essa necessidade, porém, elas possuem alto custo operacional com hardware e

software confiáveis, e com sua administração e manutenção por profissionais especializados em tecnologia. Seguindo a tendência mundial, para diminuição do custo operacional em implementar estruturas locais de tecnologia deste tipo, é contratar serviços de organizações que oferecem serviços completos de *data warehouse* em nuvem, como Oracle, IBM, Microsoft, Amazon e Google (Neto, 2021).

2.1.7 *Business Intelligence*

Business Intelligence (BI) são técnicas, métodos, ferramentas e tecnologias disponíveis para usuários analisarem dados procedentes da DW. Esses dados são de grande importância para a organização e servirão para processos decisórios dentro da mesma. Alguns usuários departamentais dentro de uma corporação têm acesso às informações para níveis decisórios. O acesso, extração e recuperação de dados de uma DW, do ponto de vista do BI, podem ser feitas das seguintes formas: Ferramentas de Consulta e Emissão de Relatórios, *Dashboards* (Painéis Digitais), Ferramentas OLAP, Ferramentas de Data Mining, DSS e EIS (Neto, 2021).

Existem diferenças entre BI e *Big Data*. No BI as informações analisadas, em geral, refletem apenas o que já ocorreu. Essas informações são extraídas do local em que foram geradas e armazenadas, e após os processos de ETL, são utilizadas para construção de relatórios e/ou *dashboards* para apresentação executiva. Uma solução BI sozinha não possui inteligência própria, ela somente reflete, de forma condensada e única, fatos passados, e se faz necessário que profissionais especialistas interpretem e tomem as decisões sobre os dados apresentados. A interpretação desses dados pode possibilitar a identificação e a análise de indicadores de desempenho e assim possam ser tomadas decisões para correções ou melhorias no fluxo operacional para obtenção de um melhor desempenho nos negócios. Já o *Big Data*, não é um novo BI, nem um BI melhorado. *Big Data* é a inclusão de ferramentas e processos de inteligência nas soluções com base em análises de grandes volumes de dados, de diversas origens e formatos e que estão em constante movimento.

BI e *Big Data* são distintos e possuem objetivos diferentes, porém estão correlacionados. *Big Data* preocupa-se menos com exatidão dos dados e tem em seu foco o processamento de grandes volumes de dados em busca de novas correlações de dados. BI não busca dados, esse nem é seu objetivo, sua função é sintetizar, sumarizar e apresentar resultados de áreas de negócios sobre os principais fatos de uma organização. BI trabalha sobre dados passados, e não procura descobrir nada de novo além dos dados que já são de domínio das áreas, não é aplicada nenhuma técnica de Mineração de Dados para se descobrir alguma correlação nova entre os dados. Já para o *Big Data*, o diferencial é mostrar caminhos e correlações de dados antes desconhecidas, não é importante saber os motivos das correlações existentes entre os dados. *Big Data* busca o quê dos fatos, não o porquê como o BI (Machado, 2018).

2.1.8 Ferramentas existentes

Várias são as ferramentas existentes no mercado para se trabalhar com *Big Data*. Abaixo, estão listadas duas delas:

a) DBeaver

DBeaver é uma ferramenta universal de gerenciamento de banco de dados para trabalhar com dados, de forma profissional. Com o DBeaver se pode manipular dados como em uma planilha normal, criar relatórios analíticos baseados em registros de diferentes armazenamentos de dados e exportar informações em um formato apropriado. Para usuários avançados de banco de dados, o DBeaver sugere um editor de SQL poderoso, muitos recursos de administração, habilidades de migração de dados e esquema, monitoramento de sessões de conexão de banco de dados. O DBeaver pronto para uso oferece suporte a mais de oitenta bancos de dados. Tendo a usabilidade como objetivo principal, o DBeaver oferece: Interface do usuário, cuidadosamente projetada e implementada; suporte de fontes de dados em nuvem; suporte para padrão de segurança organizacional; capacidade de se trabalhar com várias extensões para integração com Excel, Git e outros; grande número de recursos e suporte multiplataforma (DBeaver, 2021).

DBeaver é uma ferramenta que proporciona trabalhar, simultaneamente, com várias fontes de dados distintas, sem fazer correlação entre os dados de cada fonte. Essa forma de trabalhar com várias fontes de dados distintas está no conceito de Variedade dentro do *Big Data*.

b) *BigQuery*

O *BigQuery* é o serviço de armazenamento de dados para análise, totalmente gerenciado, em escala de petabyte e econômico do *Google Cloud*, que permite executar análises em vastos volumes de dados quase em tempo real. Com o *BigQuery* não há infraestrutura para configurar ou gerenciar, permitindo que o usuário se concentre em conseguir *Insights* significativos com o SQL padrão e aproveitar modelos de preços flexíveis em opções *on demand* e de taxa fixa (*BigQuery*, 2021).

BigQuery é uma ferramenta que proporciona trabalhar com *Big Data*, ou seja, trabalhar com grandes volumes de dados, com várias fontes de dados distintas e velocidade quase que em tempo real. Após a importação das fontes, a ferramenta disponibiliza as mesmas na estrutura de tabelas, sendo possível consultar os dados de forma separada, ou fazendo junções entre os dados dessas fontes de acordo com os relacionamentos criados entre elas, utilizando a linguagem SQL, tudo isso, de acordo com o conceito de Banco de Dados Relacional.

2.2 DADOS

Os dados são elementos que constituem a matéria-prima da informação. São definidos, também, como conhecimento bruto, ainda não devidamente tratado para prover *insights* para uma organização. Assim, os dados representam um ou mais

significados que, de forma isolada, não conseguem ainda transmitir uma mensagem clara (*know solutions*, 2019).

Já as informações são os dados devidamente tratados e analisados, produzindo conhecimento relevante. Ao contrário dos dados, elas têm significados práticos e podem ser utilizadas para reforçar o processo de tomada de decisão (*know solutions*, 2019).

Dados podem ser estruturados, semi-estruturados e não estruturados. Independente do tipo, eles ficam sempre armazenados em “arquivos” distribuídos através das redes (Neto, 2021).

2.2.1 Arquivo

É a base de processamento de um computador. Arquivos existem em vários tipos de formatos, como: textos, imagens, bancos de dados, gráficos etc. São organizados em pastas, que podem ser visualizadas utilizando o gerenciador de arquivos do sistema operacional, em discos locais, ou em discos virtuais na nuvem (Neto, 2021).

Arquivos e Processamento em *Big Data* são distribuídos e ficam espalhados em diferentes nós (computadores, discos) da rede, interconectados via rede local ou de longa distância. Trata-se de um sistema de processamento distribuído em paralelo, permitindo que o processamento seja executado em um nó mais próximo, ou mesmo subdividido para vários outros (Neto, 2021).

Abaixo, são mostrados alguns exemplos de formatos de arquivos:

a) Planilha

Como mostrada no Microsoft Excel, no LibreOffice Calc e no Google Planilhas, é um modo conveniente de armazenar informações (Oracle Brasil, 2021).

Foi originalmente projetada para único usuário, ou um pequeno número de usuários (suas características refletem isso) que não precisam fazer manipulações de dados extremamente complicadas. (Oracle Brasil, 2021).

Planilha, no caso Planilha Eletrônica, ou Folha de Cálculo, ou ainda Planilha de Cálculo, é uma implementação por meio de programas de computador, que utiliza tabelas para realização de cálculos, ou apresentação de dados. Cada tabela é formada por uma grade composta de linhas e colunas (Planilhas eletrônicas, 2017).

b) *Comma-Separated Values* (CSV)

Valores Separados por Vírgula (CSV), é um formato de arquivo que contém linhas de dados de texto compostas por colunas separadas por vírgula. O formato é útil porque é fácil de analisar e pode ser carregado e editado com qualquer programa de planilha. Como as vírgulas podem fazer parte dos dados, qualquer coluna que inclua uma vírgula, esta deve ser colocada entre aspas duplas (Fry, 2008).

c) *eXtensible Markup Language* (XML)

Linguagem de Marcação Extensível (XML) é uma linguagem de marcação, que incorpora *tags* de estrutura em seu conteúdo. Essas *tags* possibilitam permitir flexibilidade, na inclusão de números arbitrários de elementos de qualquer tamanho, em ordens variadas. Essas *tags* delineiam e identificam o conteúdo encontrado no documento (Fry, 2008).

Documentos XML são relativamente fáceis de analisar, mas, mesmo que documentos sejam projetados para facilitar a análise, é preciso ter em mente quais dados realmente são necessários no arquivo (Fry, 2008).

d) *JavaScript Object Notation* (JSON)

Notação de objetos JavaScript (JSON) é um formato para intercâmbio de dados. Um formato para intercâmbio de dados é um formato-texto usado para trocar dados entre plataformas. É um formato muito útil, pois mesmo quando essas plataformas possuem tecnologias bem diferentes, a comunicação entre elas terá um padrão que ambas conseguem “entender” (Bassett, 2019).

JSON foi criado a partir de um subconjunto do JavaScript. Conhecer essa linguagem antes de aprender JSON tem o seu valor, porém, não é pré-requisito, pois a essência de um formato de intercâmbio de dados, é ser independente de linguagem (Bassett, 2019).

A sintaxe do JSON é baseada no conceito de pares nome-valor, esse conceito é muito difundido em computação. Esses pares são conhecidos também por outros nomes: pares chave-valor, pares atributo-valor e pares campo-valor. Em um par nome-valor, inicialmente, deve-se declarar o nome, por exemplo, “animal”, em seguida deve-se declarar o valor para esse nome. Para simplificar o exemplo, o valor será um valor string, mas em pares nome-valor, em JSON, o valor também pode ser um número, um booleano, null (nulo), um array, ou um objeto. Para esse par nome-valor, cujo nome é “animal”, será usado o valor de string “cat” e por ser string, o valor deve ter aspas duplas, mas se fosse, por exemplo, número, não precisaria. Diferentemente do valor, o nome precisa sempre possuir aspas duplas ao seu redor (Bassett, 2019).

O JSON utiliza o caractere dois-pontos (:), para separar os nomes dos valores no par nome-valor. O nome está sempre à esquerda e o valor sempre à direita. O único elemento que está faltando para a sintaxe do JSON estar completa é o elemento que torna o par nome-valor, um objeto, as chaves em torno do mesmo. Com o nome, o valor, o dois-pontos e as chaves, a sintaxe do JSON para o exemplo citado está completa e fica da seguinte forma: { “animal” : “cat” } (Bassett, 2019).

2.2.2 Banco de Dados

Assim como a planilha, é um modo conveniente de armazenar informações (Oracle Brasil, 2021).

Ao contrário da planilha, é projetado para conter coleções muito maiores de informações organizadas, quantidades enormes, às vezes. Permite que vários

usuários, ao mesmo tempo, acessem e consultem com rapidez e segurança os dados, usando lógica e linguagem altamente complexas (Oracle Brasil, 2021).

Um banco de dados é uma coleção organizada de dados, ou informações, normalmente armazenadas eletronicamente em um sistema de computador. Um banco de dados é geralmente controlado por um sistema de gerenciamento de banco de dados (SGBD). Os dados e o SGBD, juntamente aos aplicativos associados a eles, são chamados de sistema de banco de dados, geralmente abreviados para apenas banco de dados (Oracle Brasil, 2021).

2.2.3.1 Banco de Dados Relacional

É uma tecnologia utilizada em larga escala em sistemas comerciais, bancários, reservas de voo, ou aplicações com dados estruturados. SQL é a linguagem de consulta orientada para estas aplicações (Neto, 2021).

SGBDR têm vantagens em dois aspectos: esquemas que permitem o controle e a validação dos dados e relacionamentos que permitem as conexões entre as diferentes tabelas (Neto, 2021).

Tabelas são objetos de banco de dados que contêm todos os dados em um banco de dados relacional. Nas tabelas, os dados são organizados de maneira lógica em um formato de linha-e-coluna semelhante ao de uma planilha. Cada linha representa um registro exclusivo e cada coluna representa um campo no registro (Documentos do SQL, 2021).

Relacionamentos ou Relações são as associações estabelecidas entre duas ou mais tabelas. As relações são baseadas em campos comuns de mais de uma tabela, envolvendo, muitas vezes, chaves primárias e estrangeiras (Suplementos do Office, 2021).

Há essencialmente três tipos de relações:

- Um-para-um: para cada registro na tabela primária, há apenas um registro na tabela estrangeira.
- Um-para-muitos: para cada registro na tabela primária, há um ou mais registros relacionados na tabela estrangeira.
- Muitos para muitos: para cada registro na tabela primária há vários registros relacionados na tabela estrangeira e para cada registro na tabela estrangeira há vários registros relacionados na tabela primária.

(Suplementos do Office, 2021).

2.2.3.2 Banco de Dados NoSQL

Aplicações de bancos de dados se destacam na consistência de esquemas de dados, podem ser escalados, mas não foram projetados para um dimensionamento infinito (Neto, 2021).

A necessidade de analisar dados em grandes volumes, de diversas fontes e formatos, fez surgir a tecnologia NoSQL. Essa tecnologia não é baseada em esquemas, regras que governam dados ou objetos. Em essência, todas as implementações NoSQL buscam manuseio em escala de grandes volumes de dados não estruturados (Neto, 2021).

Bancos de dados NoSQL podem crescer sem fim e se concentram mais em desempenho, permitindo a replicação dos dados em vários nós da rede, lendo, escrevendo e processando dados a velocidades inimagináveis, usando os paradigmas de processamento paralelo distribuído (Neto, 2021).

A tecnologia NoSQL pode ser usada em análises de dados em tempo real, como, por exemplo: em personalização de sites a partir do rastreamento do comportamento do usuário e em Internet das Coisas (IoT), a partir da telemetria de dispositivos móveis (Neto, 2021).

NoSQL permite relacionamentos aninhando documentos. Por exemplo, um documento pai poderia ter um documento filho aninhado diretamente a ele. Muitos mecanismos de consulta NoSQL suportam nativamente a capacidade de realizar consultas e associações com base em documentos complexos e aninhados (Neto, 2021).

Bancos de dados NoSQL possuem vários tipos. Os mesmos podem ser vistos abaixo:

a) *Column Database* (Banco de dados orientado por colunas)

Um banco de dados NoSQL que armazena dados em tabelas e as gerencia por colunas ao invés de linhas é chamado de sistema de gerenciamento de banco de dados colunar (SGBDC). Neste SGBDC, colunas se transformam em arquivos de dados e um dos benefícios dessa transformação é que os dados podem ser compactados, permitindo que operações como cálculos de mínimo, máximo, soma, contagem e médias sejam executadas rapidamente (Neto, 2021).

Os SGBDC podem ser auto indexados, usando menos espaço em disco do que um sistema de banco de dados relacional contendo os mesmos dados (Neto, 2021).

Um exemplo de SGBDC, é o Apache HBase, um NoSQL orientado por Colunas. (Neto, 2021).

Figura 7 - Apache HBase (Créditos: Apache Foundation)

Fonte: Neto, 2021.

b) *Key-Value Database* (Banco de dados orientado por chave/valor)

Um banco de dados orientado por chave/valor é um estrutura que armazena dados de forma a se assemelhar à estrutura de mapa ou dicionário. A chave é um identificador para um registro, ela referencia o valor desejado, já o valor pode ser do

tipo inteiro, cadeia de caracteres (*string*), estrutura de arquivos no formato JSON, ou Matriz (Neto, 2021).

Um exemplo de banco de dados orientado por chave/valor é o Apache Cassandra. Cassandra é um poderoso NoSQL. Foi originalmente desenvolvido pelo Facebook em 2008, sendo extremamente escalável e tolerante a falhas. Foi desenvolvido para resolver problemas analíticos de *Big Data*, em tempo real, envolvendo petabytes de dados (Neto, 2021).

Figura 8 - Apache Cassandra (Créditos: Apache Foundation)

Fonte: Neto, 2021.

c) *Document Database* (Banco de dados orientado por documentos)

NoSQL orientados a documentos são semelhantes aos de chave/valor. Eles organizam os documentos em coleções análogas a tabelas relacionais, e a pesquisa pode ser feita baseada também em valores, e não apenas baseada em chave (Neto, 2021).

Um exemplo de banco de dados orientado por documentos é o Apache MongoDB, que armazena dados no formato JSON de documentos como se fossem esquemas, significando que os campos podem variar de um documento para outro e a estrutura de dados podendo ser alterada ao longo do tempo. Este NoSQL é desenvolvido pela própria MongoDB Inc. e distribuído gratuitamente pela Apache Foundation (Neto, 2021).

Figura 9 - Apache MongoDB (Créditos: Apache Foundation)

Fonte: Neto, 2021.

2.2.3 Grafo

É composto por conjuntos de nós ou objetos que se conectam entre si por arestas, e que podem ser representados em forma matricial para serem processados por computadores. São utilizados para modelar “coisas” como redes rodoviárias, redes sociais, fluxo de mercadorias, etc (Neto, 2021).

Um exemplo de ferramenta baseada em grafos é o Apache Giraph. Apache Giraph é uma implementação escalável e tolerante a falhas de algoritmos de processamento de grafos em clusters Hadoop para milhares de nós computacionais. É utilizado em larga escala pelo LinkedIn, Twitter e Facebook para análises de redes sociais, com a representação de bilhões ou trilhões de conexões em conjuntos de dados para identificar popularidade, importância, localização e interesses entre os usuários (Neto, 2021).

Figura 10 - Apache Giraph (Créditos: Apache Foundation)

Fonte: Neto, 2021.

3. DESENVOLVIMENTO DO PROJETO

Este capítulo aborda o desenvolvimento prático do projeto.

3.1 OBJETIVO

O objetivo do projeto é a construção de um Processador de Consulta. Este, por sua vez, irá realizar consultas e cruzamentos de dados em fontes de dados estruturadas distintas, com base no conceito de variedade do *Big Data*. O Processador de Consulta irá facilitar a análise de informações necessárias e essenciais ao usuário, no que diz respeito ao seu negócio.

3.2 PROCESSADOR DE CONSULTA

É uma ferramenta (*software*) onde o usuário irá utilizá-la para realizar consultas e cruzamentos de dados em fontes de dados estruturadas distintas. Possui um menu principal, no qual o usuário acessa. Este menu possui outras duas opções: Fonte de Dados e Consulta e Cruzamento de Dados.

3.2.1 Menu Fonte de Dados

O menu Fonte de Dados fornece operações de CRUD (*Create*, *Read*, *Update*, *Delete*) ao usuário, que é um acrônimo para as maneiras de se operar em informação armazenada. É um mnemônico para as quatro operações básicas de armazenamento persistente. Tipicamente refere-se a operações performadas em um banco de dados ou base de dados (Mozilla, 2021).

O menu Fonte de Dados irá fornecer operações de CRUD em registros que estão armazenados na tabela FONTE_DADO, criada em um banco de dados PostgreSQL, mostrada na Figura 11.

Figura 11 - Tabela FONTE_DADO

FONTE_DADO		
IDENTIFICADOR	numeric	<pk>
NOME	character(50)	
ID_TIPO_FONTE_DADO	numeric	
HOST_BD	character(20)	
PORTA_BD	character(10)	
USUARIO_BD	character(20)	
SENHA_BD	character(50)	
BASE_DADOS_BD	character(50)	
ESQUEMA_BD	character(50)	
CAMINHO_ARQUIVO	character varying(500)	

Fonte: Próprio Autor.

A tabela FONTE_DADO possui 10 colunas que são: IDENTIFICADOR, coluna de tipo numérico e chave primária da tabela. NOME, coluna de tipo caractere, que armazena os nomes das pessoas e precisa ser preenchida. ID_TIPO_FONTE_DADO, coluna de tipo numérico, que considera a fonte de dados, se esta for um banco de dados relacional, o valor armazenado é 1, se for um arquivo, o valor armazenado é 2 e precisa ser preenchida. HOST_BD, coluna de tipo caractere e armazena o *host* do banco de dados, PORTA_BD, coluna de tipo caractere e armazena a porta de acesso ao banco de dados, BASE_DADOS_BD, coluna de tipo caractere e armazena o nome da base de dados, USUARIO_BD, coluna de tipo caractere e armazena o nome do usuário de acesso ao banco de dados, SENHA_BD, coluna de tipo caractere e armazena a senha do usuário de acesso ao banco de dados, ESQUEMA_BD, coluna de tipo caractere e armazena esquema do banco de dados e CAMINHO_ARQUIVO, coluna de tipo caractere e armazena diretório onde o arquivo se encontra.

Se a fonte de dados for um banco de dados relacional, então as colunas preenchidas, além das de preenchimento obrigatório, serão apenas as que fazem referência a banco de dados. Se a fonte de dados for um arquivo, então a coluna preenchida, além das colunas de preenchimento obrigatório, será apenas a que faz referência ao caminho do arquivo.

3.2.2 Menu Consulta e Cruzamento de Dados

Recebe o Comando de Consulta da Linguagem de Consulta Multifontes informado pelo usuário, o repassa ao Motor de Consulta e obtém de volta o resultado das consultas e dos cruzamentos, que é mostrado ao usuário.

A Linguagem de Consulta Multifontes é uma linguagem criada exclusivamente para utilização no Processador de Consulta, foi desenvolvida com base na SQL e sua definição pode ser vista no Quadro 1.

Quadro 1 - Definição da Linguagem de Consulta Multifontes

COMANDO_CONSULTA =>	SELECT SEÇÃO_CONSULTAS SEÇÃO_FONTES SEÇÃO_CRUZAMENTOS [SEÇÃO_FILTRAGEM] [SEÇÃO_ORDENAÇÃO] ;
SEÇÃO_CONSULTAS =>	CONSULTA CONSULTA [{CONSULTA}]
CONSULTA =>	(CONSULTA_FONTE_DADOS) AS NOME_RESULTADO
SEÇÃO_FONTES =>	FROM FONTE_DADOS, FONTE_DADOS [{, FONTE_DADOS}]
FONTE_DADOS =>	NOME_RESULTADO.FONTE_DADOS
SEÇÃO_CRUZAMENTOS =>	{ON CRUZAMENTO}
CRUZAMENTO =>	NOME_RESULTADO.ATRIBUTO = NOME_RESULTADO.ATRIBUTO
SEÇÃO_FILTRAGEM =>	{WHERE FILTRO}
FILTRO =>	NOME_RESULTADO.ATRIBUTO < | <= | = | >= | > VALOR_CONSTANTE
SEÇÃO_ORDENAÇÃO =>	ORDER BY ORDENAÇÃO [{, ORDENAÇÃO}]
ORDENAÇÃO =>	NOME_DO_RESULTADO.ATRIBUTO

Fonte: Próprio Autor.

O Comando de Consulta tem início na palavra reservada “SELECT” e fim no caractere válido “;”. Entre eles há várias seções.

A primeira é a SEÇÃO_CONSULTAS, onde são informadas as consultas às fontes de dados. Se a fonte de dados possuir uma linguagem própria de consulta, como por exemplo, um SGBDR, que no caso é a SQL, então será necessário informar a mesma entre os parênteses. Se a fonte de dados não possuir linguagem própria, é necessário informar apenas os nomes das colunas desejadas. A cada consulta à fonte de dados informada, deve ser informado também um nome de resultado. Este será atribuído como o nome do resultado da execução de cada fonte de dados.

A segunda é a SEÇÃO_FONTES, onde são informados os nomes das fontes de dados. Estes devem estar previamente cadastrados, conforme descrito no tópico 3.1.1. A cada fonte de dados informada, deve ser informado também um nome de

resultado, que deve ser idêntico ao nome do resultado dado a consulta à fonte de dados que será executada na fonte de dados em questão.

A terceira é a SEÇÃO_CRUZAMENTOS, onde, são informados os parâmetros de cruzamento de dados. O nome do resultado irá indicar à qual fonte de dados pertence o atributo informado. O nome do atributo não precisa ser idêntico nas fontes de dados, mas o teor dos dados referentes a ele, sim.

A quarta é a SEÇÃO_FILTRAGEM, onde são informados os filtros desejados para visualização do resultado do cruzamento. O nome do resultado irá indicar à qual fonte de dados pertence o atributo informado. E para definir como será realizada a filtragem, são utilizados caracteres de comparação, juntamente à constante desejada. Esta não é uma seção obrigatória na execução do Comando de Consulta.

A quinta é a SEÇÃO_ORDENAÇÃO, onde são informados os parâmetros de ordenação desejados para visualização do resultado do cruzamento e da filtragem (se houver). O nome do resultado irá indicar à qual fonte de dados pertence o atributo informado. A ordenação dos dados é feita em ordem crescente. Esta não é uma seção obrigatória na execução do Comando de Consulta.

As chaves, conforme mostradas no Quadro 1, marcam as partes que podem repetir, os colchetes marcam as partes opcionais.

A Linguagem de Consulta Multifontes possui palavras reservadas, que podem ser vistas no Quadro 2. Os símbolos aceitos pela linguagem podem ser vistos no Quadro 3.

Quadro 2 - Palavras Reservadas

Palavra reservada	Descrição
SELECT	Recupera dados de duas ou mais fontes de dados, utilizando a consulta à fonte de dados.
AS	Aplica um nome ao resultado a uma consulta à fonte de dados.
FROM	Indica as fontes de dados das quais recupera dados.
ON	Indica qual será o parâmetro de cruzamento das fontes de dados.
WHERE	Inclui um predicado de comparação, que restringe as linhas retornadas pela consulta. Elimina todas as linhas do conjunto de resultados onde o predicado de comparação não é avaliado como verdadeiro.
ORDER BY	Identifica quais colunas usar para classificar os dados resultantes e em que direção classificá-los (crescente ou decrescente).

Fonte: Próprio Autor.

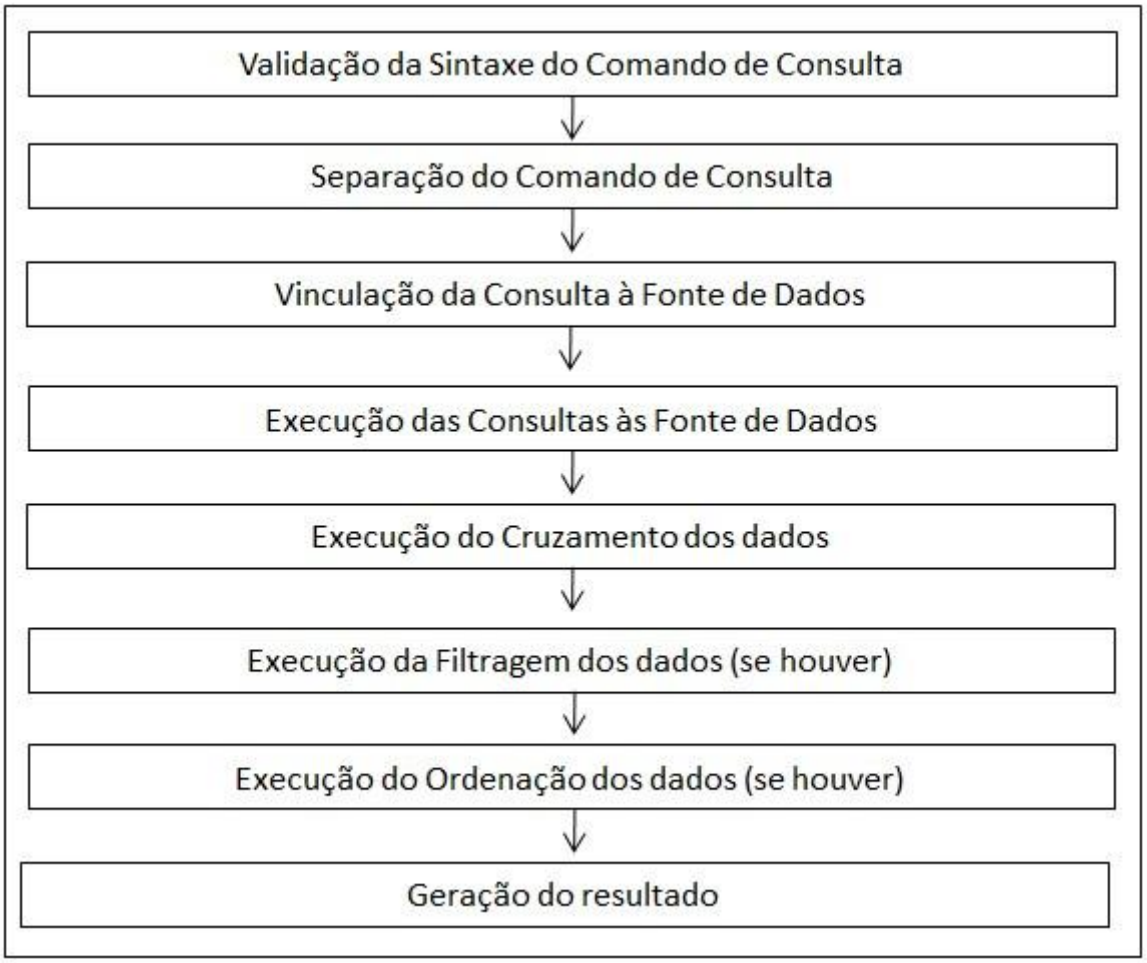
Quadro 3 - Caracteres válidos

Caractere válido	Descrição
(	Indica o início da consulta à fonte de dados.
)	Indica o fim da consulta à fonte de dados.
,	Separa as fontes de dados das quais recupera dados. Sempre deve haver o nome de uma fonte de dados do lado esquerdo da vírgula e uma do lado direito.
.	Vincula a fonte de dados e os parâmetros de cruzamento, filtragem e ordenação de dados a um resultado de uma consulta à fonte de dados.
;	Marca o fim do Comando de Consulta.
(	Indica o início da consulta à fonte de dados.

Fonte: Próprio Autor.

O Motor de Consulta recebe o Comando de Consulta e realiza várias etapas internamente para gerar o resultado das consultas e dos cruzamentos. Em cada etapa é realizada uma função específica e as mesmas podem ser vistas na Figura 12.

Figura 12 - Motor de Consulta



Fonte: Próprio Autor.

Na primeira etapa é realizada a Validação da Sintaxe do Comando de Consulta, que consiste na validação das palavras reservadas e dos caracteres válidos, nos quesitos assertividade de escrita e ordem que foram informados.

Na segunda etapa é realizada a Separação do Comando de Consulta, que consiste em separar cada seção em objetos. Cada objeto possui formato específico de acordo com cada seção em que foi baseada e o elo de ligação entre esses objetos é o nome do resultado.

Na terceira etapa é realizada a Vinculação da Consulta à Fonte de Dados, que consiste na busca da fonte de dados informada na seção fontes que, neste momento, é um objeto de fontes. Este objeto possui o nome do resultado e o nome da fonte de dados e se realmente essa fonte de dados tiver sido cadastrada, então a busca, o objeto terá também os dados de conexão para a fonte de dados.

Na quarta etapa é realizada a Execução das Consultas às Fontes de Dados, que consiste na execução das consultas às fontes de dados e obtenção do resultado. Se a fonte de dados for um SGBDR, é repassado o comando SQL informado pelo usuário para o mesmo executar. Se a fonte de dados for um arquivo, o próprio motor consulta irá ler os dados do mesmo. Independentemente da forma que é retornado o resultado da consulta pela fonte de dados, o mesmo sempre sai

desta etapa em formato de mapa, para melhor manipulação dos objetos dentro das etapas seguintes.

Na quinta etapa é realizada a Execução do Cruzamento de Dados, que consiste no cruzamento dos dados das fontes de dados informadas na seção fontes, de acordo com os parâmetros de cruzamento.

Na sexta etapa é realizada a Execução da Filtragem de Dados, que consiste na filtragem dos dados das fontes de dados resultantes do cruzamento, de acordo com os parâmetros de filtragem.

Na sétima etapa é realizada a Execução da Ordenação de Dados, que consiste no ordenação dos dados das fontes de dados resultantes do cruzamento e da filtragem (se houver), de acordo com os parâmetros de ordenação.

Na oitava etapa é realizada a Geração do Resultado, que consiste no retorno do resultado das consultas às fontes de dados, do cruzamento, da filtragem (se houver) e da ordenação (se houver), em formato de tuplas.

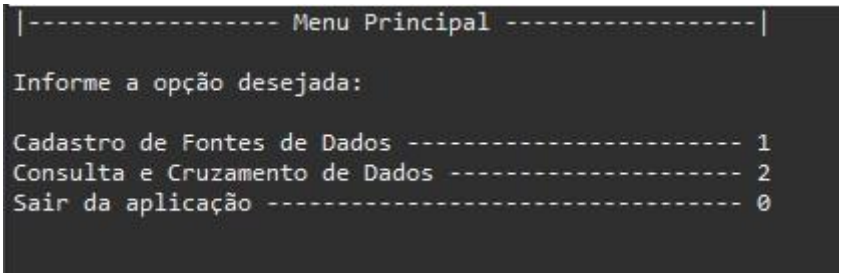
O resultado das consultas e dos cruzamentos em formato de tuplas é obtido do Motor de Consulta e mostrado na tela ao usuário.

3.3 FUNCIONAMENTO E TELAS

Um exemplo do funcionamento do Processador de Consulta com base em suas telas é mostrado abaixo:

a) O usuário acessa o Menu Principal, conforme mostrado na Figura 13.

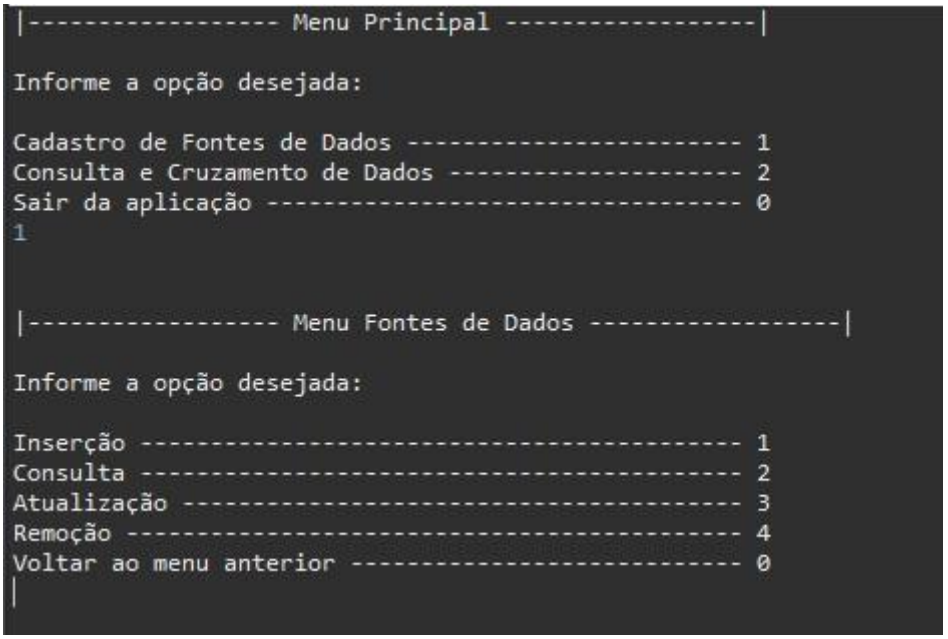
Figura 13 - Menu Principal



Fonte: Próprio Autor.

b) O usuário acessa o Menu Fonte de Dados através do Menu Principal, conforme mostrado na Figura 14.

Figura 14 - Menu Fonte de Dados



Fonte: Próprio Autor.

c) O usuário faz uma consulta às fontes de dados cadastradas, conforme mostrado na Figura 15.

Figura 15 - Menu Fonte de Dados (Consulta)

Fonte: Próprio Autor.

d) O usuário retorna ao Menu Principal e acessa o Menu Consulta e Cruzamento de Dados, conforme mostrado na Figura 16.

Figura 16 - Menu Consulta e Cruzamento de Dados

Fonte: Próprio Autor.

e) O usuário informa o Comando de Consulta, conforme mostrado na Figura 17.

Figura 17 - Menu Consulta e Cruzamento de Dados (Comando)

```
|------------------ Menu Consulta e Cruzamento de Dados ------------------|

Informe a opção desejada:

Informar Comando ---------------------------------- 1
Voltar ao menu anterior --------------------------- 0
1


Informe o Comando:


SELECT
        (
            SELECT "IDENTIFICADOR", "NOME", "CPF" FROM fd_sgbd_postgresql."PESSOA" ORDER BY "NOME"
        ) AS RST1
        (
            NOME, SEXO
        ) AS RST2

FROM RST1.FD_SGBD_PostgreSQL, RST2.FD_Arquivo_CSV_Pessoa

ON RST1.NOME = RST2.NOME

;
```

Fonte: Próprio Autor.

f) O usuário visualiza o resultado da consulta à primeira fonte de dados, conforme mostrado na Figura 18.

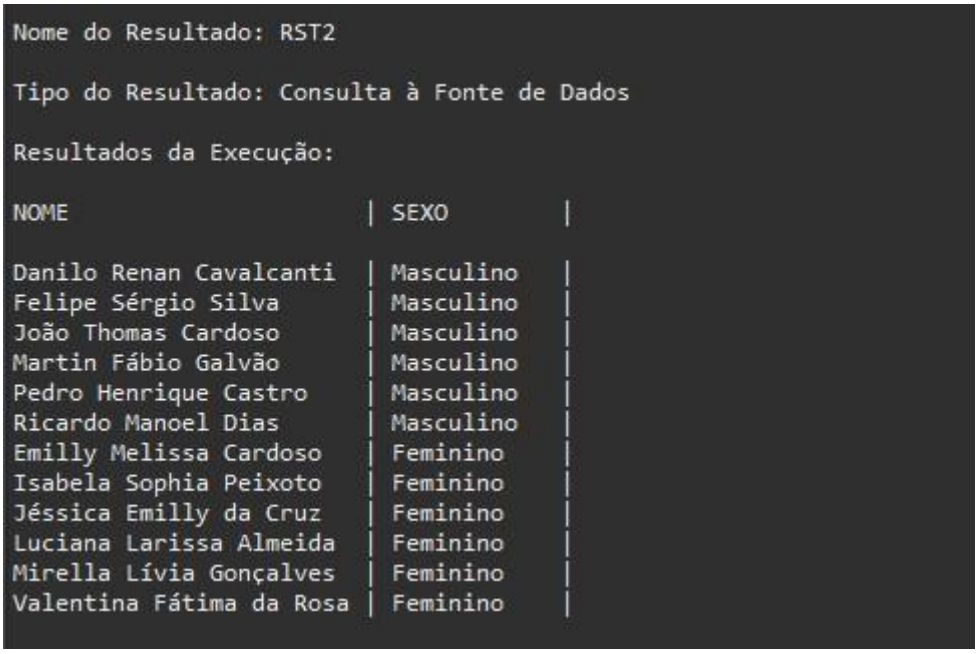
Figura 18 - Resultado 1

```
Resultado 1:


Nome do Resultado: RST1

Tipo do Resultado: Consulta à Fonte de Dados

Resultados da Execução:

IDENTIFICADOR        | NOME                    | CPF         |

1                    | Felipe Sérgio Silva     | 80213688964 |
5                    | Isabela Sophia Peixoto  | 43659031607 |
2                    | João Thomas Cardoso     | 23492374352 |
6                    | Luciana Larissa Almeida | 90897441931 |
3                    | Martin Fábio Galvão     | 51438511493 |
7                    | Mirella Lívia Gonçalves | 35049790204 |
4                    | Ricardo Manoel Dias     | 56807943229 |
8                    | Valentina Fátima da Rosa | 61316172368 |
```

Fonte: Próprio Autor.

g) O usuário visualiza o resultado da consulta à segunda fonte de dados, conforme mostrado na Figura 19.

Figura 19 - Resultado 2



```
Nome do Resultado: RST2

Tipo do Resultado: Consulta à Fonte de Dados

Resultados da Execução:

NOME                    | SEXO       |

Danilo Renan Cavalcanti | Masculino  |
Felipe Sérgio Silva     | Masculino  |
João Thomas Cardoso     | Masculino  |
Martin Fábio Galvão     | Masculino  |
Pedro Henrique Castro   | Masculino  |
Ricardo Manoel Dias     | Masculino  |
Emilly Melissa Cardoso  | Feminino   |
Isabela Sophia Peixoto  | Feminino   |
Jéssica Emilly da Cruz  | Feminino   |
Luciana Larissa Almeida | Feminino   |
Mirella Lívia Gonçalves | Feminino   |
Valentina Fátima da Rosa| Feminino   |
```

Fonte: Próprio Autor.

h) O usuário visualiza o resultado do cruzamento da primeira e da segunda fonte de dados, conforme mostrado na Figura 20.

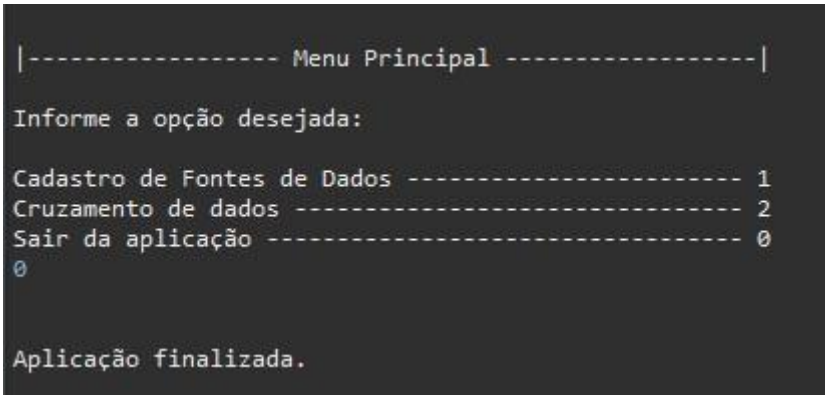
Figura 20 - Resultado 3

```
Resultado 3:


Nome do Resultado: RST1 + RST2

Tipo do Resultado: Cruzamento de Dado

Resultados da Execução:

IDENTIFICADOR | NOME                    | CPF         | NOME                    | SEXO       |

1             | Felipe Sérgio Silva     | 80213688964 | Felipe Sérgio Silva     | Masculino  |
5             | Isabela Sophia Peixoto  | 43659031607 | Isabela Sophia Peixoto  | Feminino   |
2             | João Thomas Cardoso     | 23492374352 | João Thomas Cardoso     | Masculino  |
6             | Luciana Larissa Almeida | 90897441931 | Luciana Larissa Almeida | Feminino   |
3             | Martin Fábio Galvão     | 51438511493 | Martin Fábio Galvão     | Masculino  |
7             | Mirella Lívia Gonçalves | 35049790204 | Mirella Lívia Gonçalves | Feminino   |
4             | Ricardo Manoel Dias     | 56807943229 | Ricardo Manoel Dias     | Masculino  |
8             | Valentina Fátima da Rosa| 61316172368 | Valentina Fátima da Rosa| Feminino   |
```

Fonte: Próprio Autor.

i) O usuário retorna ao Menu Principal e encerra a aplicação, conforme mostrado na Figura 21.

Figura 21 - Encerramento da aplicação



Fonte: Próprio Autor.

3.4 MATERIAIS E MÉTODOS

O processador de consulta foi construído utilizando a linguagem de programação Java, versão jdk1.8.0_271.

Conforme informado na seção 3.3.3, o processador de consulta armazena dados no SGBD PostgreSQL. A versão utilizada é a 13.2-1.

4. CONCLUSÃO E CONSIDERAÇÕES FINAIS

Este trabalho relatou sobre a definição, estrutura e dificuldades de *Big Data*, composta de cinco Vs: Velocidade, Volume, Variedade, Veracidade e Valor, além dos tipos, cruzamentos e as estruturas de dados, ferramentas de análise de dados e os detalhes do desenvolvimento e do funcionamento do Processador de Consulta. Discutiram-se os desafios de se trabalhar com uma grande variedade de formatos e estruturas, que deverão ser conciliadas para as análises necessárias, a necessidade de integrar diversas fontes de dados para poderem incluir novos tipos e estruturas de dados às fontes de dados com as quais a organização já trabalha.

O objetivo do trabalho, que foi a construção de um processador para realizar consultas e cruzamentos de dados em fontes de dados estruturadas distintas, com base no conceito de Variedade do *Big Data*, foi alcançado e teve como resultado o projeto de um Processador de Consulta, que lê uma linguagem e processa o que está descrito nela. Este processamento é a consulta e o cruzamento de fontes de dados distintas e a aplicação possui algoritmos que leem essas fontes de dados separadamente e emite um resultado padronizado.

Ainda assim há alguns desafios e limitações na ferramenta: o Processador de Consulta não possui análise de fontes de dados que determina quais campos podem ser parâmetros de cruzamento, sendo necessário que o usuário informe esses parâmetros no Comando de Consulta da Linguagem de Consulta Multifontes. Outro ponto é que o Processador de Consulta não possui análise de dados do resultado do cruzamento das fontes de dados, sendo preciso que o usuário faça as análises a partir do resultado obtido.

Apesar das limitações, este processador facilitará a análise de informações necessárias e essenciais ao usuário, no que diz respeito a um negócio, com consulta e cruzamento de dados de fontes estruturadas distintas e sugere-se como melhoria no algoritmo colocar o Processador de Consulta no padrão Strategy, para permitir a implementação de um motor de busca, de modo que ele seja independente dos tipos de fontes de dados.

5. REFERÊNCIAS

Machado, Felipe Nery Rodrigues

Big Data : o futuro dos dados e aplicações / Felipe Nery Rodrigues Machado. -- São Paulo : Érica, 2018.

Neto, Jose Antonio Ribeiro

Big Data para Executivos e Profissionais de Mercado - 2ª edição / Jose Antonio Ribeiro Neto. -- 2021.

Cocian, Luis Fernando Espinosa

Manual da linguagem C /Luis Fernando Espinosa Cocian. - Canoas: Ed. ULBRA, 2004. 500p.

Fry, Ben

Visualizing Data - Exploring and Explaining Data with the Processing Environment / Ben Fry - O'Reilly Media, Incorporated, 2008.

Bassett, Lindsay

Introdução ao JSON - Um guia para JSON que vai direto ao ponto / Lindsay Bassett - Novatec Editora, São Paulo, 2019.

Planilhas eletrônicas, 2017

O que é isso? Disponível em: <https://www.professores.uff.br/lbertini/wp-content/uploads/sites/108/2017/08/planilhas.pdf>. Acessado em: 10 de dez. de 2021

DBeaver, 2021

About DBeaver. Disponível em: <https://github.com/dbeaver/dbeaver/wiki>. Acessado em: 14 de out. de 2021

BigQuery, 2021

Documentação do BigQuery. Disponível em: <https://cloud.google.com/bigquery/docs#docs>. Acessado em: 14 de out. de 2021

Documentos do SQL, 2021

Tabelas. Disponível em: <https://docs.microsoft.com/pt-br/sql/relational-databases/tables/tables?view=sql-server-ver15>. Acessado em: 10 de dez. de 2021

Suplementos do Office, 2021

Definir relações entre tabelas usando o Access SQL. Disponível em: <https://docs.microsoft.com/pt-br/office/vba/access/concepts/structured-query-language/define-relationships-between-tables-using-access-sql>. Acessado em: 10 de dez. de 2021

Oracle Brasil, 2021

Banco de dados definido. Disponível em: <https://www.oracle.com/br/database/what-is-database/#link1>. Acessado em: 10 de dez. de 2021

Qual é a diferença entre um banco de dados e uma planilha? Disponível em: <https://www.oracle.com/br/database/what-is-database/#link4>. Acessado em: 10 de dez. de 2021

Mozilla and individual contributors, 2021

CRUD. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Glossary/CRUD>. Acessado em: 18 de dez. de 2021

Synnex Westcon-Comstor, 2021

6 MAIORES DESAFIOS EM BIG DATA ENFRENTADO PELAS EMPRESAS. Disponível em: <http://digital.br.synnex.com/pt/6-maiores-desafios-em-big-data-enfrentado-pelas-empresas>. Acessado em: 09 de out. de 2021

Bruno Rafael, 2019

SQL, NoSQL, NewSQL: Qual banco de dados usar? Disponível em: <https://blog.geekhunter.com.br/sql-nosql-newsql-qual-banco-de-dados-usar/>. Acessado em: 30 de set. de 2021

know solutions, 2019

Qual a diferença entre dado e informação? Disponível em: <https://www.knowsolution.com.br/diferenca-dado-e-informacao/>. Acessado em: 14 de dez. de 2021

Os cinco Vs do Big Data, 2017

Imagem. Disponível em: <https://blog.maxieduca.com.br/wp-content/uploads/2017/03/big-data-5vs.png>. Acessado em: 20 de out. de 2021

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
GABINETE DO REITOR

Av. Universitária, 1069 ● Setor Universitário
Caixa Postal 86 ● CEP 74605-010
Goiânia ● Goiás ● Brasil
Fone: (62) 3946.1000
www.pucgoias.edu.br ● reitoria@pucgoias.edu.br

RESOLUÇÃO n° 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O(A) estudante AUGUSTO LUIZ SANTOS QUEIROZ do Curso de Ciência da Computação, matrícula 2019.1.0028.0082-8, telefone: 62 98607-8444 e-mail augustolsq@gmail.com, na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o Trabalho de Conclusão de Curso intitulado BIG DATA: Consulta e cruzamento de dados de fontes estruturadas distintas, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial de computadores, no formato especificado (Texto (PDF); Imagem (GIF ou JPEG); Som (WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 17 de dezembro de 2021.

Assinatura do(s) autor(es): Augusto Luiz Santos Queiroz

Nome completo do autor: AUGUSTO LUIZ SANTOS QUEIROZ

Assinatura do professor-orientador: Max Gontijo

Nome completo do professor-orientador: Me. Max Gontijo de Oliveira