

Relatório Técnico

Implementação de Banco de Dados Não Relacional para o projeto
"Conexão Alimentar"

Integrantes:

ERYCK RIBEIRO

FERNANDO CLAUDIANO

FRANK LIMA

GABRIEL BARBIERI

ISABELA BATISTA MACHADO DE ALMEIDA

JOÃO AUGUSTO PACOLLA

Araras - SP

2024

Sumário

Introdução	2
Estrutura de Dados em Banco de Dados Não-Relacional	2
Conversão dos Dados Relacionais para JSON	2
Exemplo de Dados em Formato JSON	3
Descrição dos Campos JSON:	4
Vantagens da Estrutura Não-Relacional	4
Conversão dos Dados CSV para JSON	4
Conclusão	6
Consultas	6
Dashboards	10

Introdução

O sistema "Conexão Alimentar" gerencia doações de alimentos, facilitando a coleta e distribuição de produtos para diferentes locais de destino. Este relatório explora a transição da estrutura relacional para um modelo de banco de dados não-relacional, especificamente MongoDB, e fornece exemplos de como os dados podem ser armazenados em formato JSON. A escolha do MongoDB e o armazenamento de produtos em arrays são explicados, mostrando as vantagens dessa abordagem para **a flexibilidade e eficiência da aplicação**.

Estrutura de Dados em Banco de Dados Não-Relacional

Os dados, originalmente armazenados em um banco de dados relacional, são transformados para uma estrutura de documentos JSON, apropriada para o MongoDB. Esta abordagem permite maior flexibilidade e melhor alinhamento com a natureza variada e semi-estruturada das doações de alimentos.

Conversão dos Dados Relacionais para JSON

O arquivo fornecido foi processado para criar documentos JSON que podem ser facilmente armazenados e consultados em um banco de dados MongoDB ou no caso a implementação no software através da linguagem de Python, utilizando Django e Pymongo

Exemplo de Dados em Formato JSON

Os dados a seguir são convertidos de seu formato relacional (CSV) extraídos do banco de dados para documentos JSON. Cada documento representa uma doação, com produtos listados em arrays.

json

```
{  
  "nome": "João Silva",  
  "data_doacao": "2023-06-01T00:00:00Z",  
  "produtos": [  
    {"nome": "Arroz", "quantidade": 2},  
    {"nome": "Feijão", "quantidade": 1},  
    {"nome": "Óleo", "quantidade": 3}  
  ],  
  "documento": "12345678901",  
  "local_destino": "Amigos do Bem"  
}
```

json

Copiar código

```
{  
  "nome": "Maria Lima",  
  "data_doacao": "2023-06-02T00:00:00Z",  
  "produtos": [  
    {"nome": "Açúcar", "quantidade": 4},  
    {"nome": "Sal", "quantidade": 2}  
  ],  
}
```

```
"documento": "23456789012",  
"local_destino": "Banco de Alimentos - ABNE"  
}
```

Descrição dos Campos JSON:

nome: Nome do doador.

data_doacao: Data da doação no formato ISO 8601.

produtos: Array de objetos, onde cada objeto representa um produto com seu nome e quantidade.

documento: Documento de identificação do doador.

local_destino: Local onde a doação será destinada.

Vantagens da Estrutura Não-Relacional

A transição para uma estrutura de banco de dados não-relacional, utilizando MongoDB, oferece várias vantagens, bem como:

Flexibilidade de Esquema: Tendo diferentes doações podem conter diferentes números e tipos de produtos sem a necessidade de modificar o esquema do banco de dados.

Escalabilidade MongoDB: Suporta escalabilidade horizontal, permitindo a expansão eficiente da capacidade de armazenamento e processamento.

Facilidade de Uso e Integração: O formato JSON é natural para dados semi-estruturados e facilita a integração com muitas aplicações web e APIs.

Conversão dos Dados CSV para JSON

Para demonstrar a conversão dos dados, os registros no arquivo CSV foram processados e convertidos em JSON. Abaixo estão os passos realizados para a conversão:

Carregar o arquivo CSV.

Transformar cada linha em um documento JSON.

Salvar os documentos JSON resultantes.

Código Python utilizado para a conversão:

```
import pandas as pd
```

```
import json
```

```
# Carregar o arquivo CSV
```

```
file_path = "/mnt/data/Conexao_alimentar.doacoes1.csv"
```

```
doacoes_df = pd.read_csv(file_path)
```

```
# Função para converter uma linha de DataFrame em JSON
```

```
def convert_row_to_json(row):
```

```
    produtos = []
```

```
    for i in range(1, 6):
```

```
        produto_nome = row[f'produto_{i}']
```

```
        if pd.notna(produto_nome) and produto_nome != '-':
```

```
            produto_qtd = int(row[f'qtd_{i}'])
```

```
            produtos.append({"nome": produto_nome, "quantidade": produto_qtd})
```

```
    doacao_json = {
```

```
        "nome": row['nome'],
```

```
        "data_doacao": row['data_doacao'],
```

```
        "produtos": produtos,
```

```
        "documento": row['documento'],
```

```
        "local_destino": row['local_destino']
```

```
    }
```

```
    return doacao_json
```

Aplicar a função a cada linha e criar uma lista de JSONs

```
doacoes_json = doacoes_df.apply(convert_row_to_json, axis=1).tolist()
```

Salvar o resultado em um arquivo JSON

```
output_path = "/mnt/data/doacoes_conexao_alimentar.json"
```

```
with open(output_path, 'w') as json_file:
```

```
json.dump(doacoes_json, json_file, indent=4)
```

Conclusão

A transição do banco de dados relacional para uma estrutura não-relacional, utilizando MongoDB e armazenamento de dados em formato JSON, **melhora a flexibilidade e a eficiência** do sistema "Conexão Alimentar". Esta abordagem é ideal para gerenciar dados de doações variáveis e **facilita a escalabilidade e a integração com outras plataformas**.

A escolha de armazenar produtos em arrays reflete a necessidade de manter uma estrutura **simples e direta para a gestão das doações**, permitindo uma **fácil manipulação e consulta** dos dados. O uso de MongoDB como banco de dados não-relacional proporciona um **armazenamento eficiente e flexível**, alinhado com as necessidades do sistema.

Consultas

Consulta 1- Quantidade geral de registros de doações no banco.

```
db.doacoes.count()
```

```
}  
> db.doacoes.count()  
< DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.  
< 235  
Conexao_alimentar>
```

Consulta 2 - Relação das 3 pessoas que mais doaram.

```
db.doacoes.aggregate([  
    { $group: { _id: "$nome", total_doador: { $sum: 1 } } },  
    { $sort: { total_doador: -1 } },  
    { $limit: 3 }  
]);
```

```

< {
  _id: 'Maikon Gino',
  total_doado: 99
}
{
  _id: 'Grace Garcia',
  total_doado: 6
}
{
  _id: 'Grace Smith',
  total_doado: 4
}

```

Consulta 3 -Relação dos 3 produtos com maior quantidade no banco

```

db.doacoes.aggregate([
  { $unwind: "$produtos" },
  { $group: { _id: "$produtos.nome", total_quantidade: { $sum: "$produtos.quantidade" } } },
  { $sort: { total_quantidade: -1 } },
  { $limit: 3 }
]);

```

```

< {
  _id: 'Molho de tomate',
  total_quantidade: 828
}
{
  _id: 'Macarrão',
  total_quantidade: 815
}
{
  _id: 'Ervilha',
  total_quantidade: 619
}

```

Consulta 4 - Relação dos 3 produtos com menor quantidade no banco

```
db.doacoes.aggregate([
  { $unwind: "$produtos" },
  { $group: { _id: "$produtos.nome", total_quantidade: { $sum: "$produtos.quantidade" } } },
  { $sort: { total_quantidade: 1 } },
  { $limit: 3 }
]);
```

```
< {
  _id: 'Sal',
  total_quantidade: 1
}
{
  _id: 'Café',
  total_quantidade: 1
}
{
  _id: 'Óleo de milho',
  total_quantidade: 1
}
```

Consulta 5- Data em que houve maior quantidade de doação.

```
db.doacoes.aggregate([
  { $group: { _id: { $dateToString: { format: "%Y-%m-%d", date: "$data_doacao" } },
total_doacoes: { $sum: 1 } } },
  { $sort: { total_doacoes: -1 } },
  { $limit: 1 }
]);
```

```
{
  _id: '2024-06-18',
  total_doacoes: 200
}
```


Consulta 6: Relação de todas as datas em que houve doação e a quantidade de cada produto

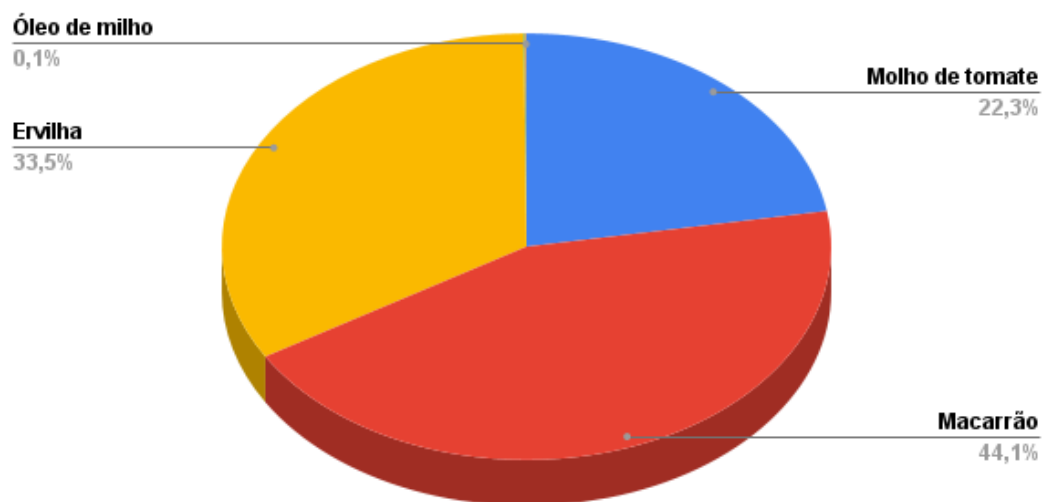
```
db.doacoes.aggregate([
  {
    $group: {
      _id: { $dateToString: { format: "%Y-%m-%d", date: "$data_doacao" } },
      total_registros: { $sum: 1 }
    }
  },
  {
    $sort: { _id: 1 } // Ordena por data crescente
  }
]);
```

```
< {
  _id: '2024-06-13',
  total_registros: 15
}
{
  _id: '2024-06-14',
  total_registros: 18
}
{
  _id: '2024-06-15',
  total_registros: 2
}
{
  _id: '2024-06-18',
  total_registros: 200
}
```

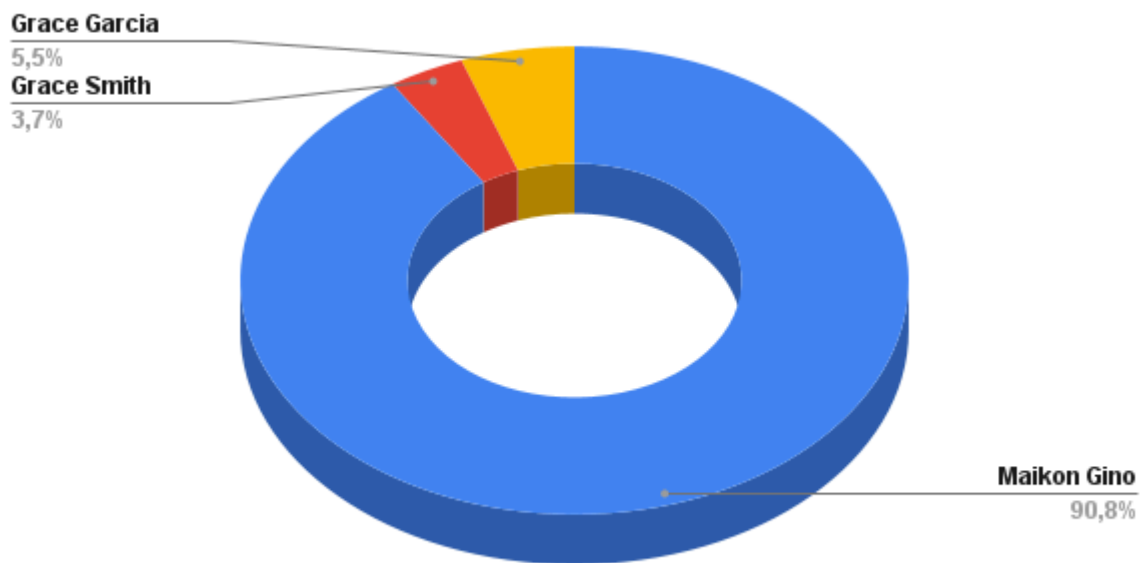
Gráfico:

Dashboards

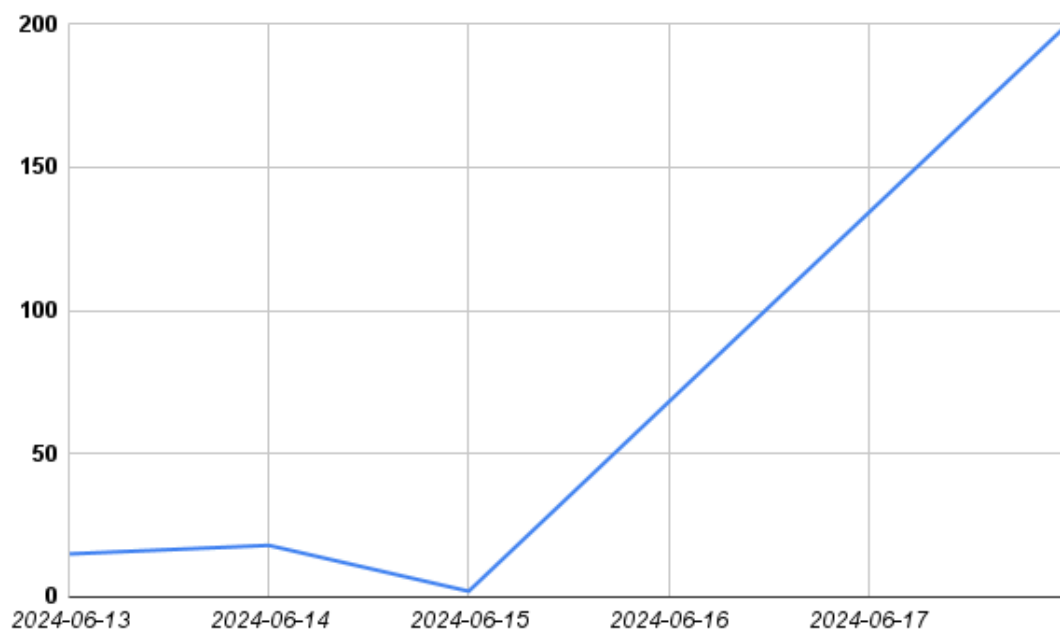
Relação dos 3 produtos mais doados com os 3 menos doados:



Relação das 3 pessoas com mais registros no banco:



Relação das datas e quantidades de registro em cada uma delas.



2024-06-13	15
------------	----

2024-06-14	18
2024-06-15	2
2024-06-18	200

Relação dos locais mais doados:

Local 1	Amparaí
Local 2	Amigos do Bem
Local 3	Banco de Alimentos - ABNE
Local 4	Cozinha - Tem Gente com Fome
Local 5	ActionAid