



INSTITUTO FEDERAL

Brasília

Campus Brasília

TECNOLOGIA EM SISTEMAS PARA INTERNET

Giovana de Oliveira Rocha

Jadilson Nascimento da Cruz Junior

Leandro Júnio de Sousa Gomes

Tatiana Dutra Ramos Casado Melo

RELATÓRIO DE PRÁTICA INTEGRADA DE CIÊNCIA DE DADOS E INTERNET DAS COISAS

Brasília - DF

11/01/2023

Sumário

1. Objetivos	3
2. Descrição do problema	4
3. Desenvolvimento	5
3.1 Instalando o MongoDB no Colab:	5
3.2 Análise dos dados e Gráficos	9
4. Considerações finais	14
Referências	15

1. Objetivos

Armazenar os dados coletados na etapa anterior em um Banco de dados, utilizando o MongoDB para o gerenciamento e preservação dos dados. Em seguida, analisar e ilustrar os dados em forma gráfica, assim como descrevê-los.

2. Descrição do problema

As principais dificuldades encontradas foram a conexão do banco de dados com o Colab, a inserção de dados nas coleções do banco de dados, a não familiaridade com a plataforma do MongoDB e na criação dos gráficos.

3. Desenvolvimento

A primeira parte do desenvolvimento se baseou na instalação, criação do banco de dados e armazenamento dos dados no MongoDB.

3.1 Instalando o MongoDB no Colab:

Primeiramente realizamos a instalação do dnspython é um kit de ferramentas DNS para Python. Ele pode ser usado para consultas, transferências de zona, atualizações dinâmicas.

```
[ ] !pip install dnspython  
  
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/  
Requirement already satisfied: dnspython in /usr/local/lib/python3.8/dist-packages (2.2.1)
```

Figura 1: Instalando o dnspython no colab.

Após a instalação, foram realizadas importações de Bibliotecas necessárias, com isso realizamos a inserção dos arquivos coletados no Arduino e transformando-os em Dicionários pelo Python, sendo a forma necessária para adicioná-los ao MongoDB a fim de imprimir os dados para averiguação.

```
1 #Importando Bibliotecas  
2 import pandas as pd  
3 from pandas import DataFrame  
4  
5 #Lendo CSV  
6 dados_normal = pd.read_csv('FluxoNormal - FluxoNormal.csv')  
7 dados_Um_Furo = pd.read_csv('FluxoUmFuro - FluxoUmFuro.csv')  
8 dados_Dois_Furos = pd.read_csv('FluxoDoisFuros - FluxoDoisFuros.csv')  
9  
10 #Convertendo para Dicionarios  
11 dados_normal_to_dict = dados_normal.to_dict(orient= "records")  
12 dados_Um_Furo_to_dict = dados_Um_Furo.to_dict(orient= "records")  
13 dados_Dois_Furos_to_dict = dados_Dois_Furos.to_dict(orient= "records")  
14  
15 print(dados_normal_to_dict)  
16 print(dados_Um_Furo_to_dict)  
17 print(dados_Dois_Furos_to_dict)
```

Figura 2: Transformações dos arquivos csv em Dicionários para implementação no MongoDB.

A Saída referente a figura 2 mostrando em seu resultado a conversão do csv para o dicionário é demonstrado a seguir:

```
[{'Quantidade': 0, 'Sensor': 'Sensor 1', 'mL': 7.43}, {'Quantidade': 1, 'Sensor': 'Sensor 2', 'mL': 7.29},  
[{'Quantidade': 0, 'Sensor': 'Sensor 1', 'mL': 1.08}, {'Quantidade': 1, 'Sensor': 'Sensor 2', 'mL': 0.13},  
[{'Quantidade': 0, 'Sensor': 'Sensor 1', 'mL': 6.89}, {'Quantidade': 1, 'Sensor': 'Sensor 2', 'mL': 4.32},
```

Figura 3: Resultado da conversão de csv para dicionário.

Foi realizado o início da conexão com o MongoDB, aqui foi coletada a String de conexão no MongoDB, conforme ilustrado abaixo:

Connect to Cluster0

✓ Setup connection security ✓ Choose a connection method Connect

1 Select your driver and version

DRIVER VERSION

Python 3.11 or later

2 Add your connection string into your application code

☒ Include full driver code example

```
client = pymongo.MongoClient("mongodb+srv://LJGT:  
<password>@cluster0.fostvjg.mongodb.net/?retryWrites=true&w=majority")  
db = client.test
```

Replace **<password>** with the password for the **LJGT** user. Ensure any option params are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back Close

Figura 4: String de conexão

Com isso teve-se o início ao desenvolvimento na conexão adicionando a string da figura 4 e foi adicionado a uma variável na qual realizou-se a conexão conforme ilustrado na linha 9 da figura 5:

```

1 import datetime      # Imports datetime library
2 import pymongo       # Importando a Biblioteca do Mongol
3 from pymongo import MongoClient  # Importação Da biblioteca MongoClient
4 import json
5 # Adicionando a URL do MongoDB
6 uri = 'mongodb+srv://LJGT:ifbedu@cluster0.fostvjg.mongodb.net/?retryWrites=true&w=majority'
7
8 # Iniciando a Conexão dos clientes ao servidor MongoDB
9 client = MongoClient( uri )

```

Figura 5: Conexão do Colab ao MongoDB

Após a conexão foi realizado a criação do Banco de dados denominado “BancodeDados” apresentado na linha 2 da figura 6 a foi criado três coleções(Tabelas) cada uma respectivamente referente a um arquivo csv.

Procedendo com o desenvolvimento, foi adicionado individualmente os dados convertidos apresentados na figura 3 dentro de cada coleção criada para o próprio.

```

1 # Criando o BD
2 BancodeDados= client["Controle_de_Fluxo"]
3
4 # Criando Colecao
5 colecao_Normal = BancodeDados["Fluxo_Normal"]
6 colecao_Um_Furo = BancodeDados["Fluxo_Um_Furo"]
7 colecao_Dois_Furos = BancodeDados["Fluxo_Dois_Furos"]
8
9 #Adicionando dados Convertidos na Coleção
10 colecao_Normal_adicionada = colecao_Normal.insert_many(dados_normal_to_dict)
11 colecao_Um_Furo_adicionadas = colecao_Um_Furo.insert_many(dados_Um_Furo_to_dict)
12 colecao_Dois_Furosadicionadas = colecao_Dois_Furos.insert_many(dados_Dois_Furos_to_dict)
13 print(colecao_Normal_adicionada.inserted_ids)
14 print(colecao_Um_Furo_adicionadas.inserted_ids)
15 print(colecao_Dois_Furosadicionadas.inserted_ids)

```

Figura 6: Realizando a criação do banco de dados e adicionando coleções e valores.

```

[ObjectId('63bec2f9b0e295082cf9e9cb'), ObjectId('63bec2f9b0e295082cf9e9cc'), ObjectId('63bec2f9b0e295082cf9e9cd'),
ObjectId('63bec2fbb0e295082cf9ebc1'), ObjectId('63bec2fbb0e295082cf9ebc2'), ObjectId('63bec2fbb0e295082cf9ebc3'),
ObjectId('63bec2fdb0e295082cf9edb7'), ObjectId('63bec2fdb0e295082cf9edb8'), ObjectId('63bec2fdb0e295082cf9edb9'),

```

Figura 7: Resultado das inserções dentro das coleção

O Banco de dados e suas coleções só serão criadas de fato após a inserção dos dados, com isso foi realizado os testes após a sua inserção.

```
1 #Teste de Conexão
2 BancodeDados.stats
```

```
Collection(Database(MongoClient(host=['ac-dmkg8wm-shard-00-01.fostvjg.mongodb.net:27017', 'ac-dmkg8wm-shard-00-02.fostvjg.mongodb.net:27017', 'ac-dmkg8wm-shard-00-00.fostvjg.mongodb.net:27017'], document_class=dict, tz_aware=False, connect=True, retrywrites=True, w='majority', authsource='admin', replicaset='atlas-lka2bo-shard-0', tls=True), 'Controle_de_Fluxo'), 'stats')
```

Figura 8: Verificando conexão do Banco de dados.

```
1 #Listando as Coleções do Banco de dados
2 print(BancodeDados.list_collection_names())
3 #colecão.drop()
```

```
['Fluxo_Dois_Furos', 'Fluxo_Um_Furo', 'Fluxo_Normal']
```

Figura 9: Listando as coleções criadas.

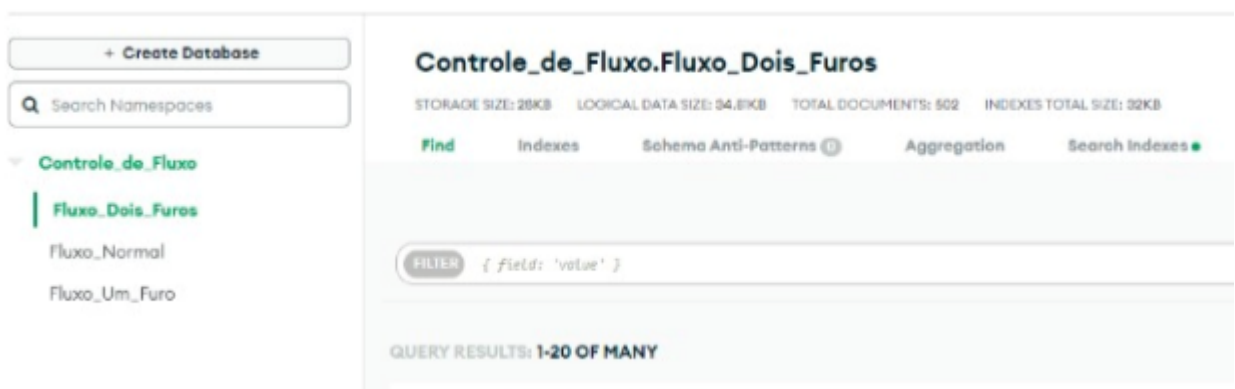


Figura 10: MongoDB com três coleções adicionadas

Link Colab:

https://colab.research.google.com/drive/1L-n06kz_c0SfXMf-ZKeb4Spedi_7iuKg?usp=sharing

Link do GitHub:

<https://github.com/infocbra/pratica-integrada-cd-e-ic-2022-2-ljtq>

3.2 Análise dos dados e Gráficos

Usando o banco de dados MongoDB para a análise dos dados e usando uma de suas funcionalidades: CHARTS para a criação dos gráficos comparativos dos dados.

Os gráficos foram construídos analisando os 50 (cinquenta) primeiros dados de cada arquivo. Para a melhor visualização dos gráficos é indicado o acesso ao GitHub inserido anteriormente, dentro da pasta “3Sprint”.

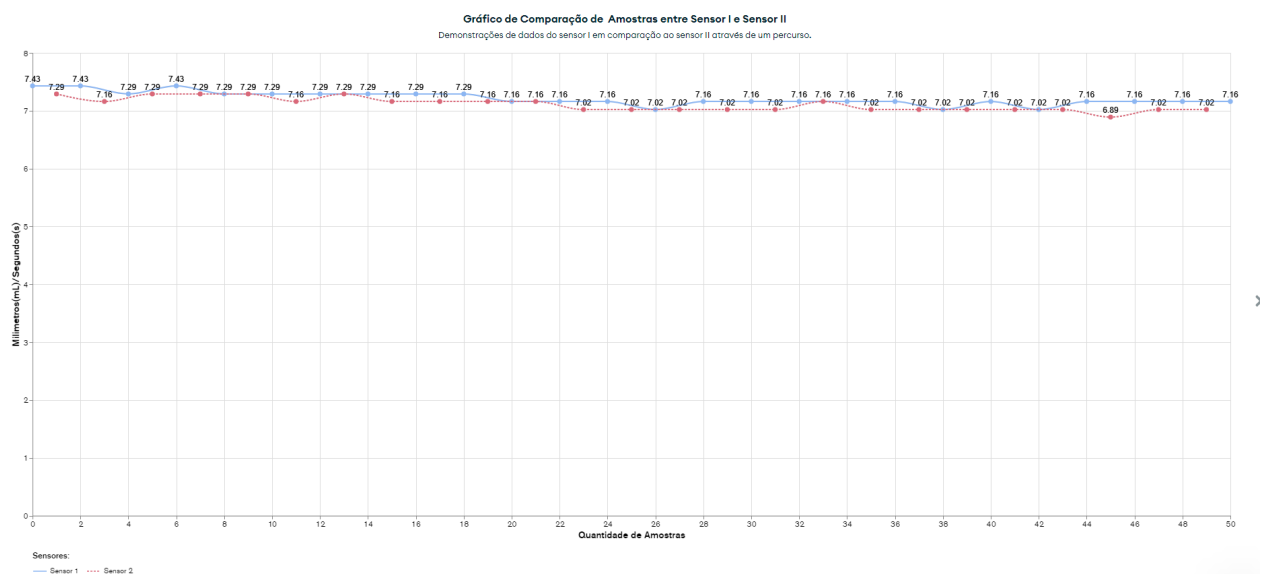


Gráfico 1: Gráfico comparativo de amostras com fluxo normal.

No gráfico 1 está disposto a comparação entre sensores e suas variações, leitura de fluxo máximo e mínimo de cada sensor.

Sensor	Máximo	Mínimo
1	7.43 mL	7.02 mL
2	7.29 mL	6.89 mL

Tabela 1 : Leitura Máxima e Mínimas do Gráfico 1

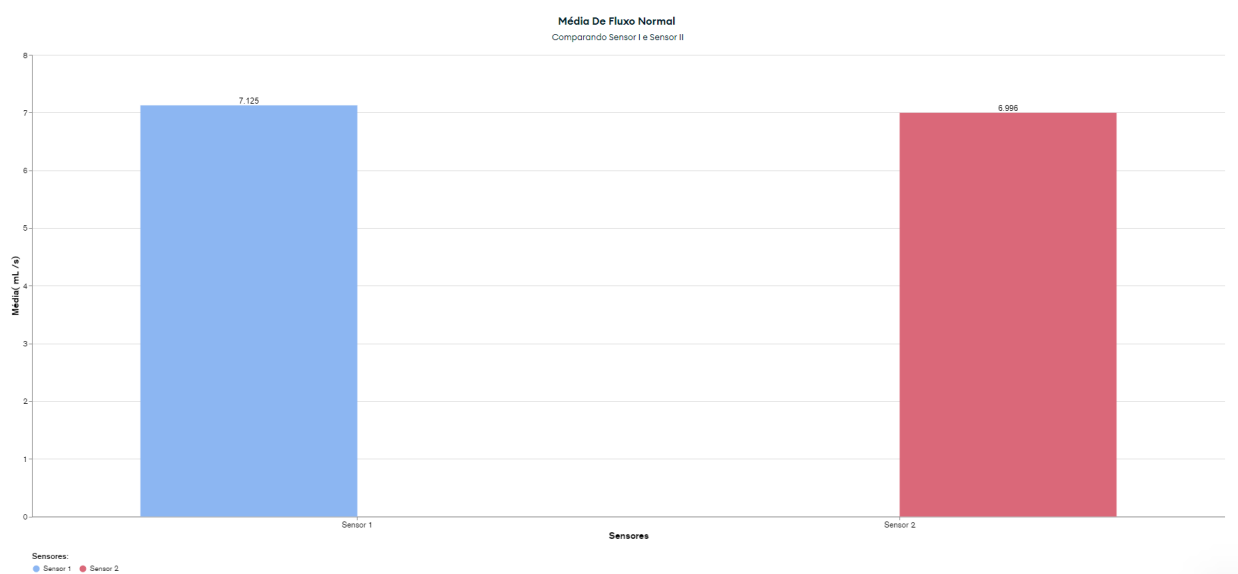


Gráfico 2: Gráfico comparativo de média entre sensores com fluxo normal.

No gráfico 2 está a média em mL por segundo de água de cada sensor com o fluxo normal.

Sensor 1:	7.125 mL
Sensor 2:	6.996 mL

Tabela 2: Leitura média do Gráfico 2

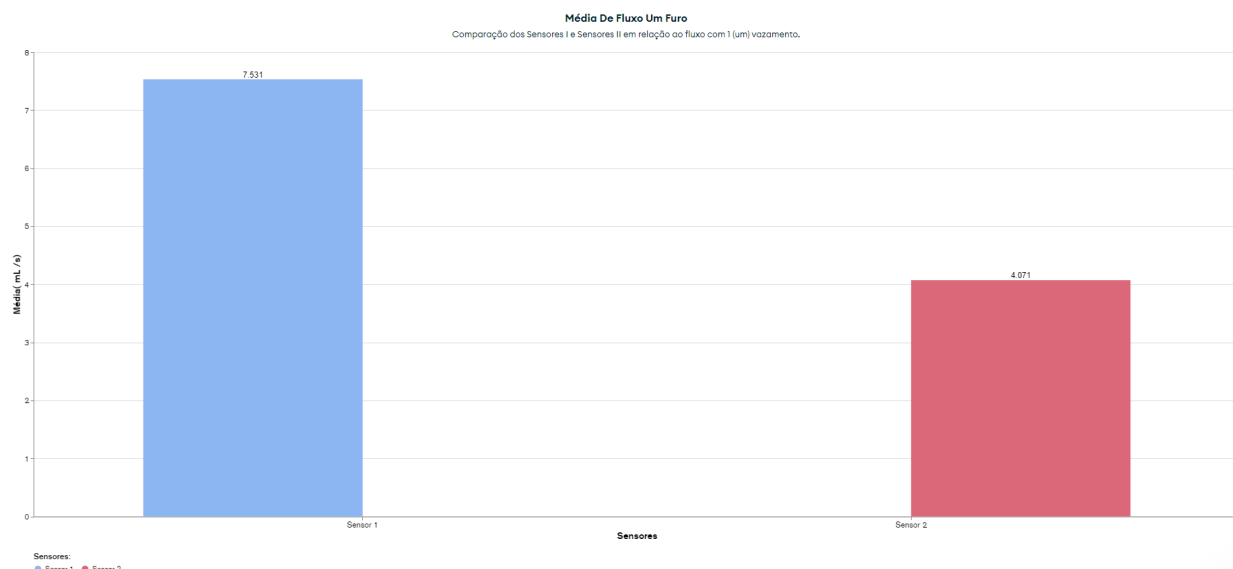


Gráfico 4: Gráfico comparativo de média entre sensores com 1(um) furo.

No gráfico 4 está a média em mL por segundo de água de cada sensor um 1(um) furo.

Sensor 1:	7.531 mL
Sensor 2:	4.701 mL

Tabela 4: Leitura média do Gráfico 4

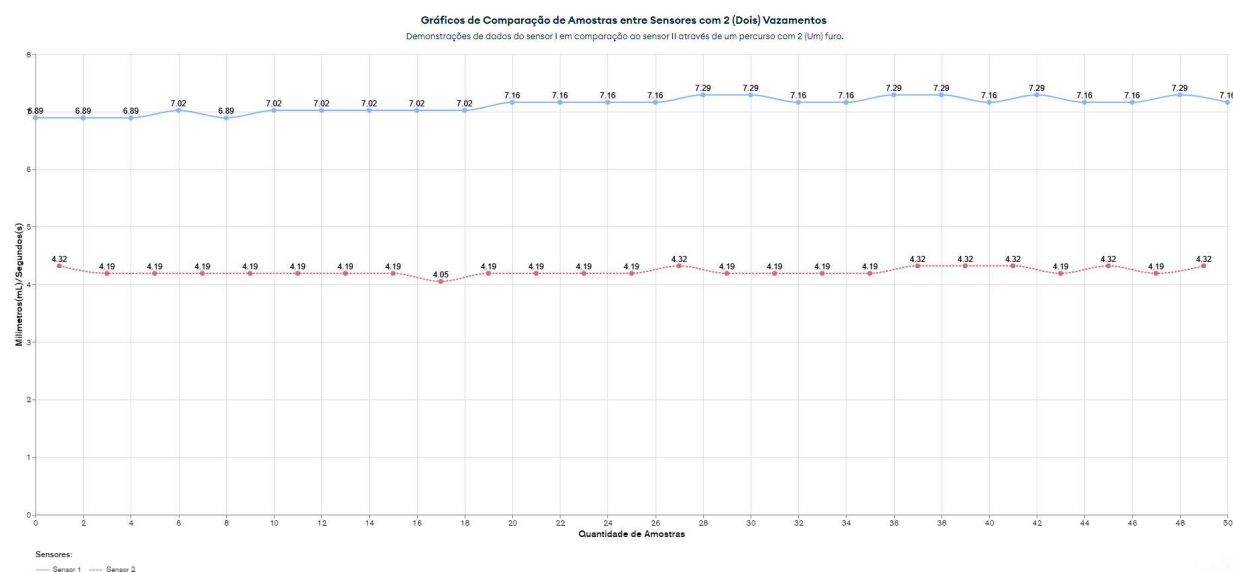


Gráfico 5 : Gráfico comparativo de amostras com 2 (dois) furos.

No gráfico 5 está disposto a comparação entre sensores e suas variações, leitura de fluxo máximo e mínimo de cada sensor, com 2 (dois) furos (evasão) de água.

Sensor	Máximo	Mínimo
1	7.29 mL	6.89 mL
2	4.32 mL	4.05 mL

Tabela 5: Leitura Máxima e Mínimas do Gráfico 5

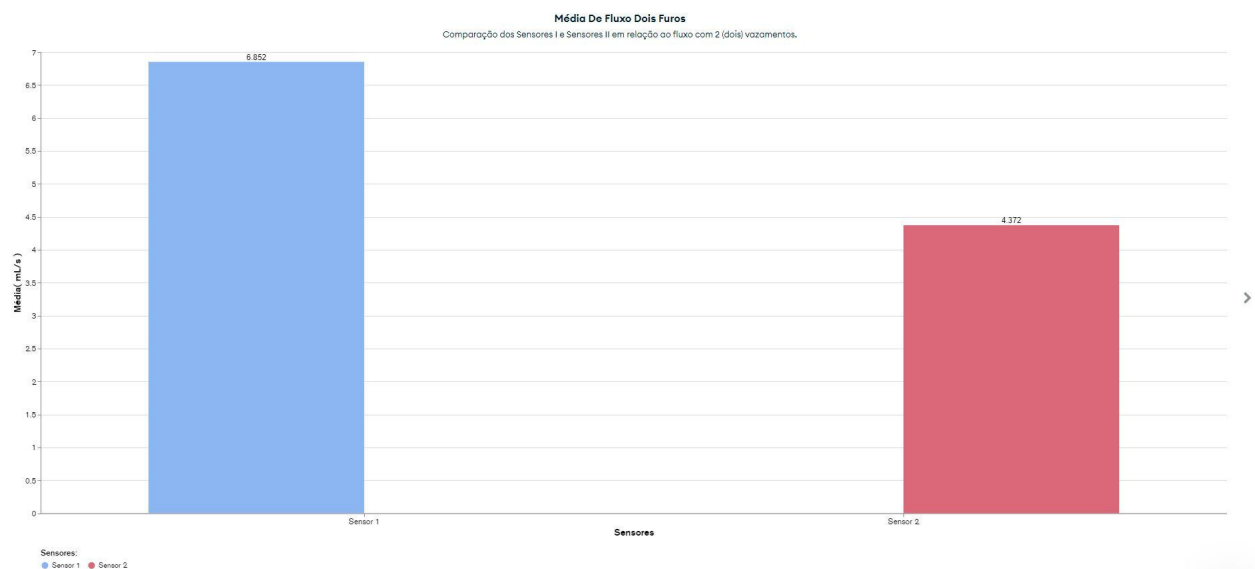


Gráfico 6 : Gráfico comparativo de média entre sensores com 2(dois) furos.

No gráfico 6 está a média em mL por segundo de água de cada sensor com 2 (dois) furos.

Sensor 1:	6.852 mL
Sensor 2:	4.372mL

Tabela 6: Leitura média do Gráfico 6.

4. Considerações finais

Na terceira splint tivemos dificuldade na familiaridade com a plataforma do MongoDB, assim como na organização de reuniões da equipe e divisão nas etapas da respectiva splint.

Referências

Eletrogate. **Guia Prático do Sensor de Fluxo de Água**. blog.eletrogate, 2022. Disponível em: <https://blog.eletrogate.com/sensor-de-fluxo-de-agua/>. Acesso em: 11/12/2022.

Mongodb. **MongoDB Charts**. mongodb.com, 2021. Disponível em: <https://www.mongodb.com/docs/charts/>. Acesso em: 11/12/2022.