APACHE NIFI - BASIC DATAFLOW

https://nifi.apache.org/nifi-docs

Ứng dụng hỗ trợ tìm kiếm vé máy bay và tra cứu giá

Mục tiêu:

- Hiểu và áp dụng quy trình ETL (Extract, Transform, Load) trong việc thu thập và xử lý dữ liệu.
- Sử dụng công cụ Apache NiFi để xây dựng pipeline tự động hóa việc thu thập, xử lý, và lưu trữ dữ liệu từ nguồn trực tuyến.
- Lưu trữ dữ liệu vào MongoDB để hỗ trợ việc tìm kiếm và tra cứu thông tin chuyến bay.
- Thực hành triển khai các công nghệ container hóa, cụ thể là Docker, để quản lý môi trường làm việc.

Nội dung bài lab:

Trong bài lab này, bạn sẽ xây dựng một pipeline data chuẩn bị dữ liệu cho ứng dụng hỗ trợ tìm kiếm và tra cứu giá vé máy bay, có thể phân tích biến động giá vé từ nguồn dữ liệu trực tuyến.

1. **Thu thập dữ liệu**:
    - Nguồn dữ liệu: https://www.vietnamairport.vn/thong-tin-lich-bay
    - Dữ liệu cần thu thập bao gồm các trường:
        - **scheduled_time**: Giờ dự kiến.
        - **estimated_time**: Giờ ước tính.
        - **route**: Lộ trình bay (ví dụ: VCA-HAN).
        - **flight_no**: Số hiệu chuyến bay.
        - **carrier**: Hãng hàng không (ví dụ: VJ).

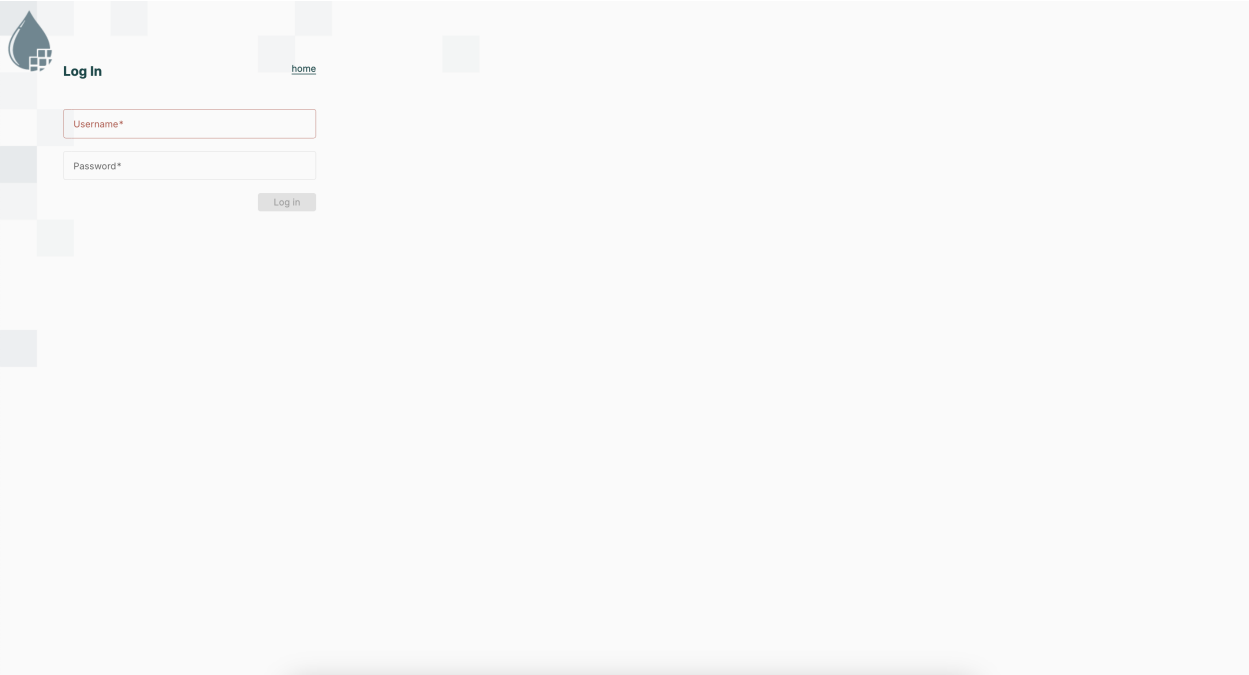
    - **cki_row**: Hàng làm thủ tục.
    - **gate**: Cổng ra máy bay.
    - **terminal**: Sảnh terminal.
    - **status**: Trạng thái chuyến bay.
    - crawled_time: Giờ thu thập.
2. **Xử lý và lưu trữ dữ liệu**:
    - Dữ liệu thu thập được cần được xử lý và lưu vào **MongoDB** để phục vụ việc tìm kiếm và tra cứu sau này.
3. **Công cụ và công nghệ sử dụng**:
    - **Apache NiFi**: Dùng để tạo quy trình ETL tự động hóa từ thu thập đến lưu trữ dữ liệu.
    - **Docker và docker compose**: Sử dụng Docker Image của NiFi và MongoDB để thiết lập môi trường làm việc với docker compose.
    - **MongoDB**: Là cơ sở dữ liệu lưu trữ dữ liệu đã thu thập.
4. **Yêu cầu triển khai**:
    - Tạo một pipeline ETL trên NiFi thực hiện các bước:
        - **Extract**: Thu thập dữ liệu từ nguồn (https://www.vietnamairport.vn/thong-tin-lich-bay)
        - **Transform**: Xử lý và chuẩn hóa dữ liệu để phù hợp với yêu cầu (Thêm cột crawled_time)
        - **Load**: Lưu dữ liệu vào MongoDB
    - Sử dụng Docker Compose để chạy NiFi và MongoDB.

Kết quả mong đợi:

- Hoàn thành pipeline ETL trên NiFi và xác minh dữ liệu được lưu đúng định dạng trong MongoDB.
- Có thể tìm kiếm và tra cứu thông tin chuyến bay dựa trên các tiêu chí như giờ dự kiến, số hiệu chuyến bay, lộ trình, v.v.

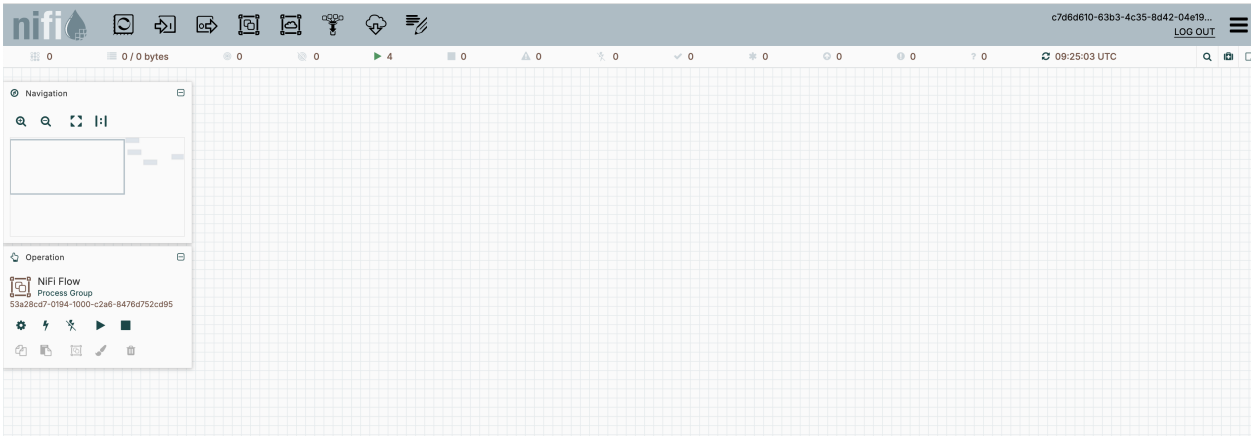
1. Setup NIFI and MongoDB

- https://hub.docker.com/r/apache/nifi
- https://hub.docker.com/_/mongo
- Ôn tập bài cũ - Sau khi hoàn thành cài đặt và khởi chạy ứng dụng, truy cập đường dẫn: https://localhost:8443/nifi/



- Login với tài khoản và mật khẩu được sinh ra

```
docker logs <container_nifi> | grep Generated
```



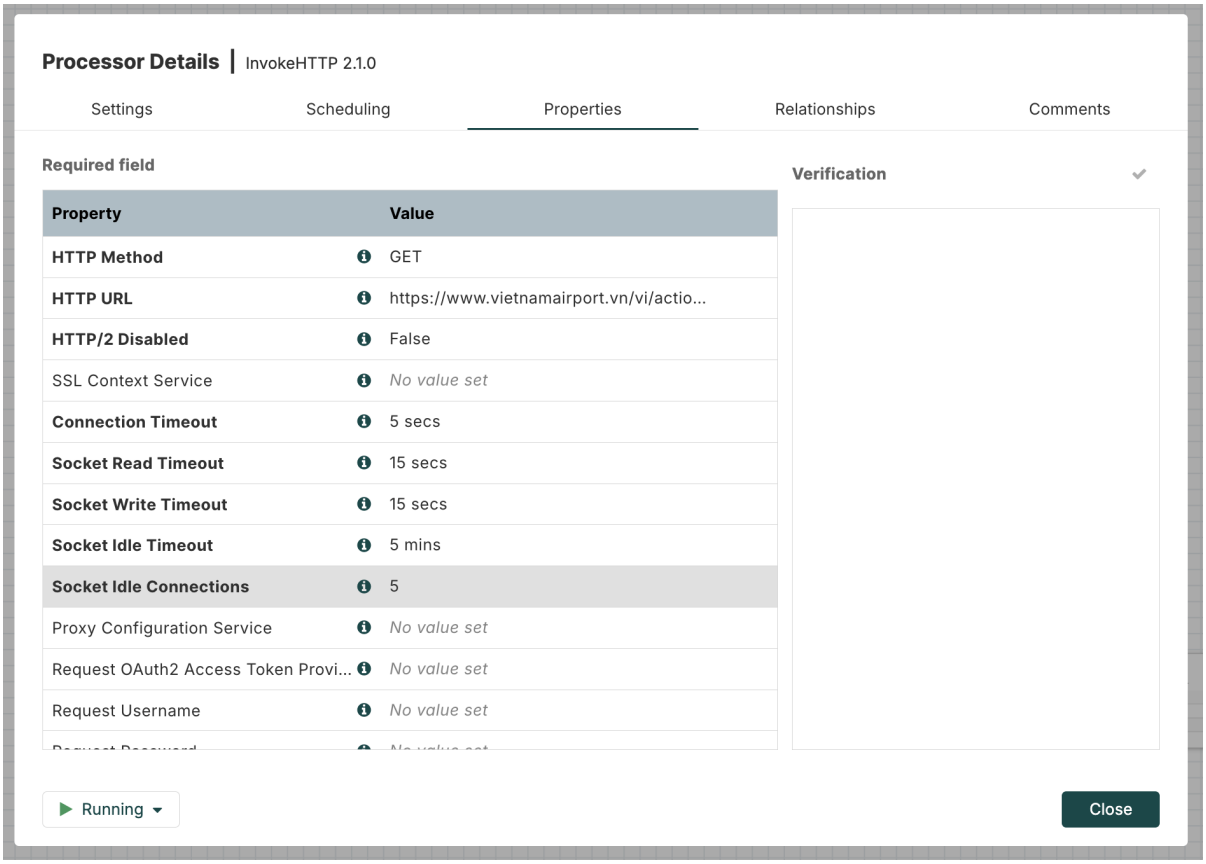
2. Xây dựng data pipeline

User Guide: https://nifi.apache.org/docs/nifi-docs/html/user-guide.html

Expression Language: https://nifi.apache.org/docs/nifi-docs/html/expression-language-guide.html

Thu thập dữ liệu - InvokeHTTP

Sử dụng EL (Expression Language):
https://www.vietnamairport.vn/vi/action/flight_plan_filter?airport=&flight_date=${now():format('yyyy-MM-dd')}&flight_no=&to_airport=&carrier=



Xử lý dữ liệu

UpdateRecord

Cấu hình service

Controller Service Details — JsonRecordSetWriter 2.1.0

Settings | Properties | Comments

Required field

Verification

Property	Value
Schema Write Strategy	Do Not Write Schema
Schema Cache	*No value set*
Schema Access Strategy	Inherit Record Schema
Date Format	*No value set*
Time Format	*No value set*
Timestamp Format	*No value set*
Pretty Print JSON	false
Suppress Null Values	Never Suppress
Allow Scientific Notation	false
Output Grouping	Array
Compression Format	none

Controller Service Details — JsonTreeReader 2.1.0

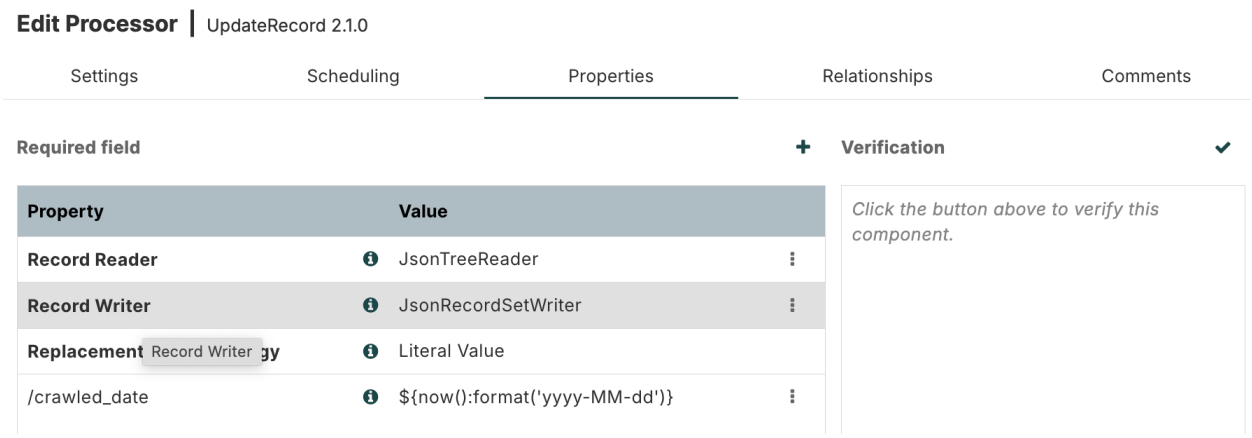
Settings | Properties | Comments

Required field

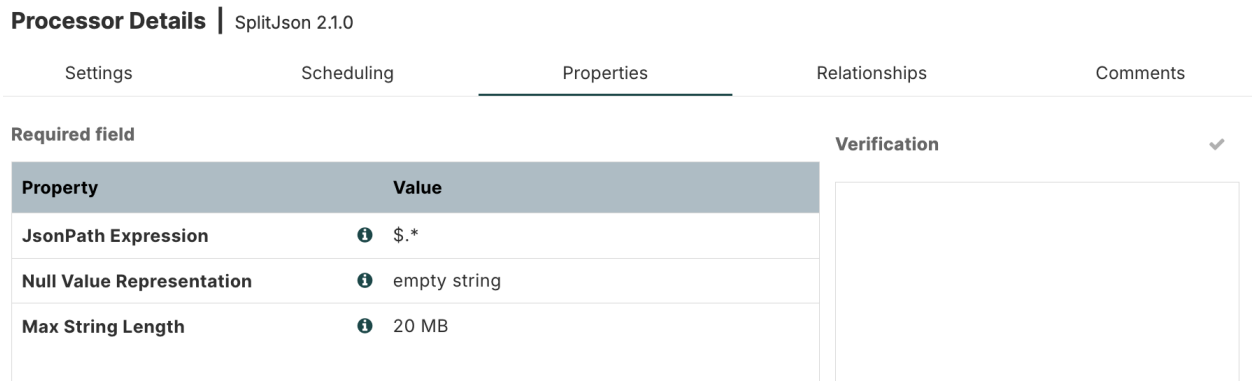
Verification

Property	Value
Schema Access Strategy	Infer Schema
Schema Inference Cache	*No value set*
Starting Field Strategy	Root Node
Max String Length	20 MB
Allow Comments	false
Date Format	*No value set*
Time Format	*No value set*
Timestamp Format	*No value set*

Cấu hình Processor

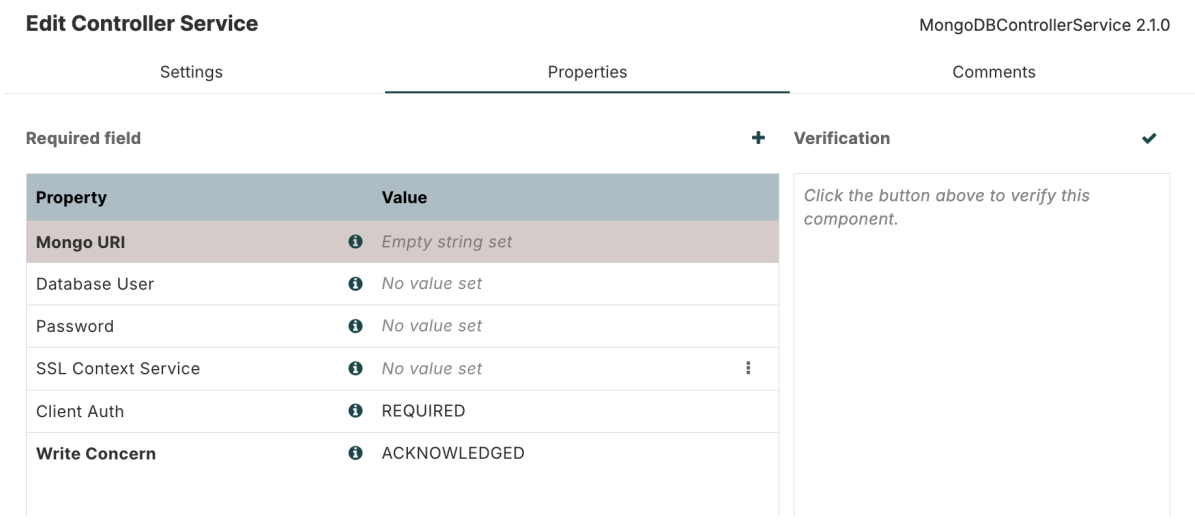


SplitJson

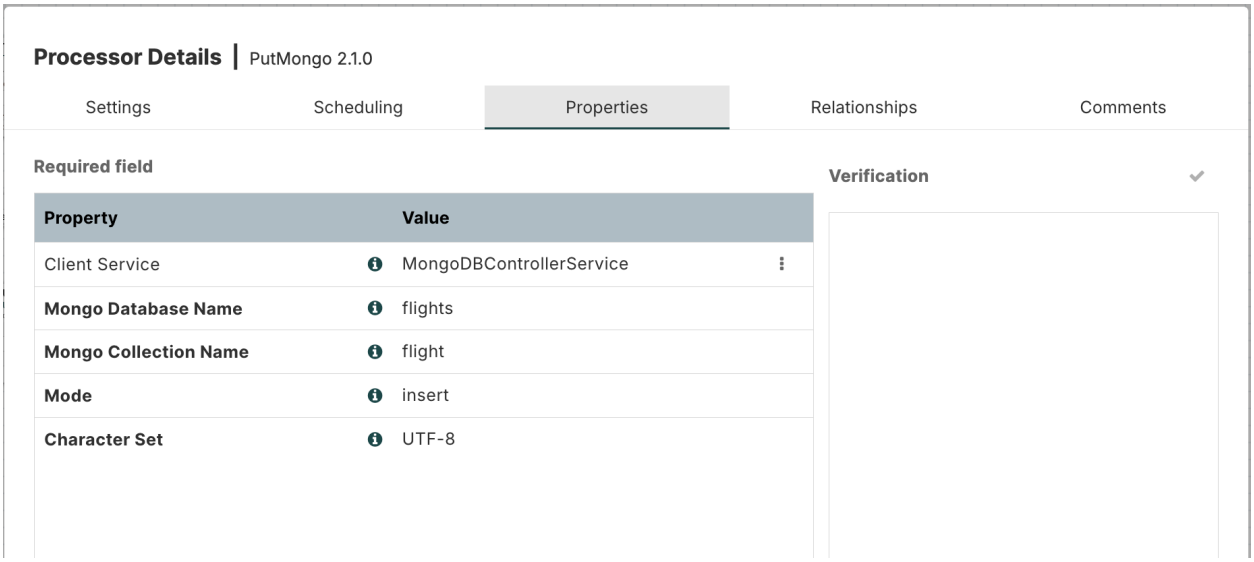


Lưu trữ dữ liệu - PutMongo

- Cấu hình service

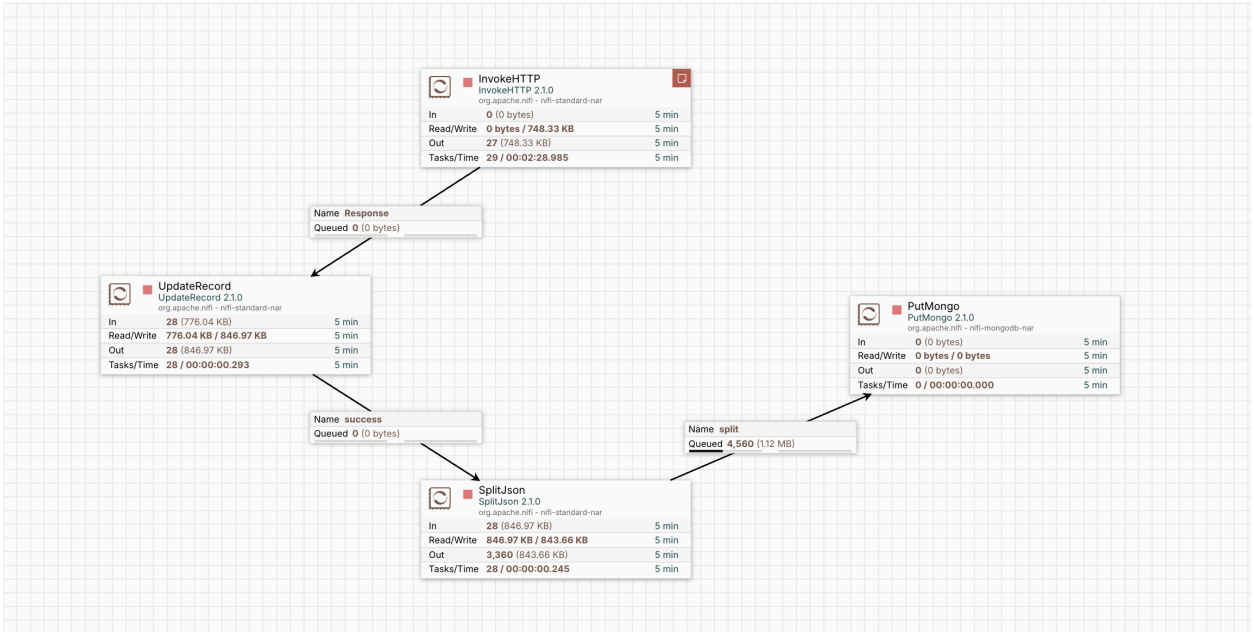


- Cấu hình Processor

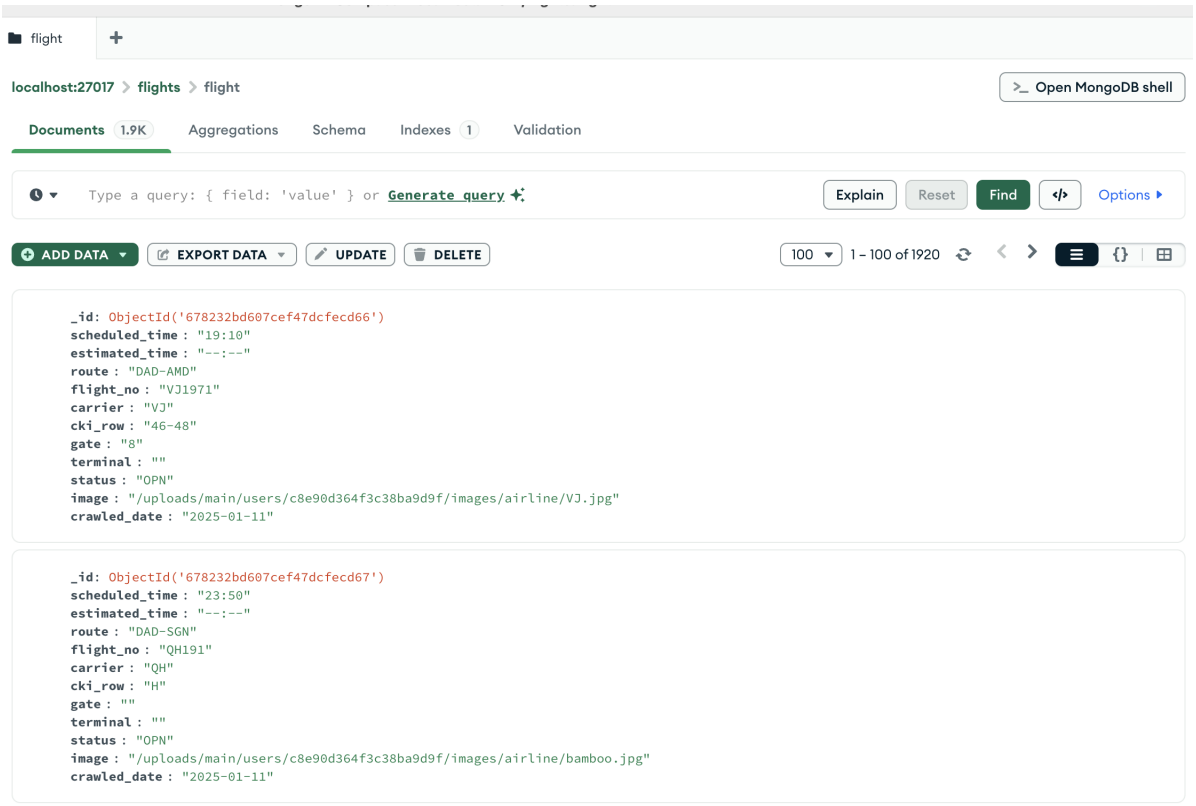


Kết nối các processor thành pipeline

Tạo các kết nối giữa các processor để hoàn thành pipeline



- Sau khi run, kiểm tra dữ liệu trong mogoDb



Clean up

```
docker compose down --volumes --rmi all --remove-orphans
```