TRƯỜNG ĐẠI HỌC NÔNG LÂM TP.HCM



**BÁO CÁO CUỐI KỲ MÔN HỌC DATA WAREHOUSE**

**ĐỀ TÀI TỶ GIÁ MẮT KÍNH**

GVHD: Th.S Nguyễn Đức Công Song

SV thực hiện:

Đặng Cao Hoàng Tuấn - 21130599

Nguyễn Kim Thịnh - 21130545

Table of content

[1. Current DW Architecture Overview 3](#_Toc184654342)

[2. Functional Specification 4](#_Toc184654343)

[2.1. Impact Analysis 4](#_Toc184654344)

[2.2. Assumptions: 4](#_Toc184654345)

[3. Exception Handling 5](#_Toc184654346)

[4. How to run job manually 6](#_Toc184654347)

[4.1. Tạo biến môi trường chứa mail nhận thông báo của chương trình 6](#_Toc184654348)

[4.2. Chạy 1 file duy thực hiện các tất cả giai đoạn 1 lúc 6](#_Toc184654349)

[4.3. Chạy riêng lẻ từng giai đoạn: Extract file 6](#_Toc184654350)

[4.4. Chạy riêng lẻ từng giai đoạn: Load to staging 7](#_Toc184654351)

[4.5. Chạy riêng lẻ từng giai đoạn: Load to dw 7](#_Toc184654352)

[5. Scheduling 7](#_Toc184654353)

[6. Logical Workflow 14](#_Toc184654354)

[6.1. Extract file process 14](#_Toc184654355)

[6.2. Load to staging process 19](#_Toc184654356)

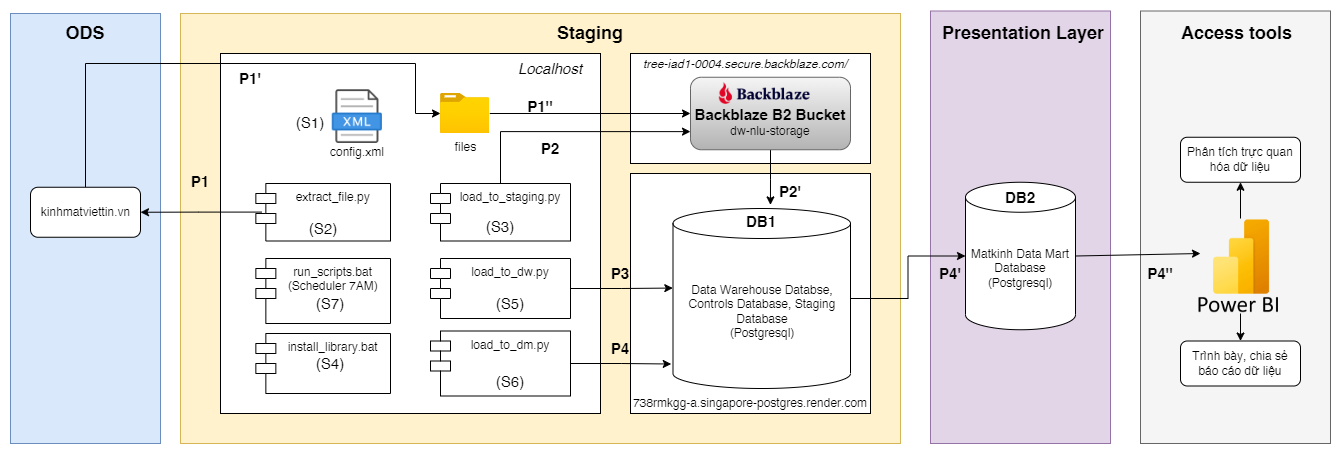
[6.3. Load to dw process 23](#_Toc184654357)

[7. Database Structure 27](#_Toc184654358)

[8. Data feed specification 31](#_Toc184654359)

[9. Script setup tables in Postgres Database 32](#_Toc184654360)

## Current DW Architecture Overview



Description:

(P1): Quá trình Extract dữ liệu:

- P1: Tiến hành cào dữ liệu từ nguồn trang web tương ứng file config.

- P1’: File .csv sau khi cào được lưu vào folder trên máy local theo file config chuẩn bị đưa lên bucket lưu trữ file tập trung trên B2.

- P1’’: Đưa file .csv lên bucket dw-nlu-storage từ file config.xml (S1) theo file config và ghi log trên Database dw (DB1).

(P2): Quá trình load dữ liệu vào staging trong Database dw (DB1)

- P2: Load file config.xml (S1) kết nối lấy url file .csv tương ứng file config.

- P2’: Import file vào bảng staging tương ứng file config.

(P3): Quá trình load dữ liệu vào Data Warehouse dw (DB1)

- P3: Insert into từ dữ liệu, đồng thời transform dữ liệu từ bảng staging vào dw..

(P4): Quá trình load dữ liệu lên Presentation Layer

- P4’: Dữ liệu Dim, Fact được đến Database Data Mart (DB2)

- P4’’: Data trên Data Mart (DB2) được đưa đến PowerBi để phân tích, báo cáo.

(S1): File config.xml chứa thông tin để kết nối các Database Server và Cloud Storage Server B2.

(S2): Script thực hiện process extract dữ liệu P1.

(S3): Script thực hiện process load dữ liệu vào staging P2.

(S4): Script giúp kiểm tra và cài đặt các thư viện cần thiết, chạy 1 lần đầu duy nhất khi bắt đầu với dự án.

(S5): Script thực hiện process load dữ liệu vào dw P3.

(S6): Script thực hiện process load dữ liệu vào dm P4.

(S7): Script thực hiện lần lượt các process, thứ tự lần lượt: P1 (Extract file) 🡪 P2 (Load to staging) 🡪 P3 (Load to dw) 🡪 P4 (Load to dm). Script được lập lịch chạy tự động với task scheduler hằng ngày vào 7AM.

(DB1): Database postgresql dw chứa các bảng file\_logs, file\_config, staging, dw.

(DB2): Database postgresql dm chứa các bảng dim, fact, report.

## Functional Specification

### 2.1. Impact Analysis

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Criteria** | **Implementer** | **Note** |
| 1 | Extract file | Cả nhóm | Quá trình cào dữ liệu từ trang web nguồn về file .csv, sau đó upload file lên backblaze b2. Ghi nhận log |
| 2 | Load to staging area | Nguyễn Kim Thịnh | Quá trình truy cập file .csv tương ứng từ B2, insert dữ liệu vào bảng staging, đồng thời transform dữ liệu trong database dw |
| 3 | Load to Data Warehouse core | Đặng Cao Hoàng Tuấn | Quá trình chuyển dữ liệu từ staging sang dw |

### 2.2. Assumptions

**Extract file module:**

* Người thực hiện:
* Thực hiện khi chạy script (S2) extract\_file.py, lập lịch task scheduler trên localhost chạy 6AM hằng ngày và thực hiện ghi log.
* Datasources:

+ Nguồn trang web lấy dữ liệu theo file config trong bảng file\_config trong Database dw (DB1): *dpg-ct2jaspu0jms738rmkgg-a.singapore-postgres.render.com.*

+ File .csv được lưu tạm ở localhost sau đó được lưu trữ ở B2 Bucket dw-nlu-storage: *tree-iad1-0004.secure.backblaze.com/*

+ File dữ liệu được lưu dưới dạng .csv.

+ Tên file theo cấu trúc: **data\_{id config}\_{table\_staging\_destiantion}\_YYYY-MM-DD\_{source}.csv**

**Load to staging module:**

* Người thực hiện:
* Thực hiện khi chạy script (S3) load\_to\_staging.py, lập lịch task scheduler trên localhost chạy 8AM hằng ngày và thực hiện ghi log.
* Datasources:

+ File .csv được lưu trữ ở B2 Bucket dw-nlu-storage: *tree-iad1-0004.secure.backblaze.com/file/dw-nlu-storage/*

+ File dữ liệu được lưu dưới dạng .csv.

+ Bảng Staging tương ứng file config được lưu trong Database dw (DB1): *dpg-ct2jaspu0jms738rmkgg-a.singapore-postgres.render.com.*

**Load to dw module:**

* Người thực hiện:
* Thực hiện khi chạy script (S3) load\_to\_dw.py, lập lịch task scheduler trên localhost chạy 10AM hằng ngày và thực hiện ghi log.
* Datasources:

+ Bảng staging tương ứng id config được lưu trong Database staging (DB1): *dpg-ct2jaspu0jms738rmkgg-a.singapore-postgres.render.com.*

+ Bảng dw được lưu trong Database dw (DB1): *dpg-ct2jaspu0jms738rmkgg-a.singapore-postgres.render.com.*

## Exception Handling

* **Normal Condition Checking**

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Criteria** | **Value** | **Action** |
| 1 | Process status | **RUNNING** | Đang trong quá trình chạy tiến trình |
| 2 | File status | **ES** (Extract Successfully) | Có thể sẵn sàng load to staging |
| **LS** (Load to staging Successfully) | Có thể sẵn sàng load to Data Warehouse dw |
| **LWS** (Load to dw successfully) | Có thể sẵn sàng load to Data mart |
| **EF** (Extract Fail) | Lỗi trong quá trình cào dữ liệu, extract file lỗi |

* **Exception/Error**

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Type** | **Exception** | **Handling** |
| 1 | Process error  (System interrupt) | **[Extract]**  Lỗi trong quá trình cào dữ liệu  *(Mất kết nối, website chặn)* | Gửi mail thông báo. Kiểm tra kết nối. Chạy lại script extract\_file.py |
| **[Load to staging]**  Không có file .csv trên B2  *(Lỗi extract file hoặc lỗi kết nối B2)* | Gửi mail thông báo.  Thử chạy lại extract\_file.py |
| **[Có tiền trình đã/đang chạy]** | Gửi mail thông báo |
| 2 | Database connection error | **[Lỗi kết nối Database]**  *(Mất kết nối Internet)* | Gửi mail thông báo |

## How to run job manually

### 4.1. Tạo biến môi trường chứa mail nhận thông báo của chương trình

* Giả sử email muốn nhận thông báo của bạn là [your\_mail@gmail.com](mailto:your_mail@gmail.com)
* Mở cmd chạy lệnh:

setx MY\_EMAIL\_DW\_VAR “[your\_mail@gmail.com](mailto:your_mail@gmail.com)”

### 4.2. Chạy 1 file duy thực hiện các tất cả giai đoạn 1 lúc

Thứ tự lần lượt: P1 (Extract file) 🡪 P2 (Load to staging) 🡪 P3 (Load to dw) 🡪 P4 (Load to dm). Script được lập lịch chạy tự động với task scheduler hằng ngày vào 7AM.

* Xét điều kiện máy đã có python và kết nối internet
* Mở cmd di chuyển đến thư mục chứa các file script
* Chạy file install\_library.bat nếu lần đầu chạy dự án.

|  |
| --- |
| #1. Kết nối đến database Controls (Postgres)  # Thông tin kết nối trong file config.xml  #2. Sử dụng câu query để kiểm tra file config vừa chọn file config  SELECT \*  FROM file\_config  #3. Chạy file run\_scripts.bat với 3 tham số <id\_config> <địa chỉ cụ thể file config.xml> <Ngày> (Optional, mặc định là ngày hôm nay, định dạng YYYY-MM-DD)  run\_scripts.bat 1 E:\Documents\NLU\DataWarehouse\project\config.xml 2024-12-12 |

### 4.3. Chạy riêng lẻ từng giai đoạn: Extract file

* Xét điều kiện máy đã có python và kết nối internet
* Mở cmd di chuyển đến thư mục chứa các file script
* Chạy file install\_library.bat nếu lần đầu chạy dự án.

|  |
| --- |
| #1. Kết nối đến database Controls (Postgres)  # Thông tin kết nối trong file config.xml  #2. Sử dụng câu query để kiểm tra file config vừa chọn file config  SELECT \*  FROM file\_config  #3. Chạy file extract\_file.py với 3 tham số <id\_config> <địa chỉ cụ thể file config.xml> <Ngày> (Optional, mặc định là ngày hôm nay, định dạng YYYY-MM-DD)  python extract\_file.py 1 E:\Documents\NLU\DataWarehouse\project\config.xml 2024-12-12 |

### 4.4. Chạy riêng lẻ từng giai đoạn: Load to staging

* Xét điều kiện máy đã có python và kết nối internet
* Mở cmd di chuyển đến thư mục chứa các file script
* Chạy file install\_library.bat nếu lần đầu chạy dự án.

|  |
| --- |
| #1. Kết nối đến database Controls (Postgres)  # Thông tin kết nối trong file config.xml  #2. Sử dụng câu query để kiểm tra file config vừa chọn file config  SELECT \*  FROM file\_config  #3. Chạy file load\_to\_staging.py với 3 tham số <id\_config> <địa chỉ cụ thể file config.xml> <Ngày> (Optional, mặc định là ngày hôm nay, định dạng YYYY-MM-DD)  python load\_to\_staging.py 1 E:\Documents\NLU\DataWarehouse\project\config.xml |

### 4.5. Chạy riêng lẻ từng giai đoạn: Load to dw

* Xét điều kiện máy đã có python và kết nối internet
* Mở cmd di chuyển đến thư mục chứa các file script
* Chạy file install\_library.bat nếu lần đầu chạy dự án.

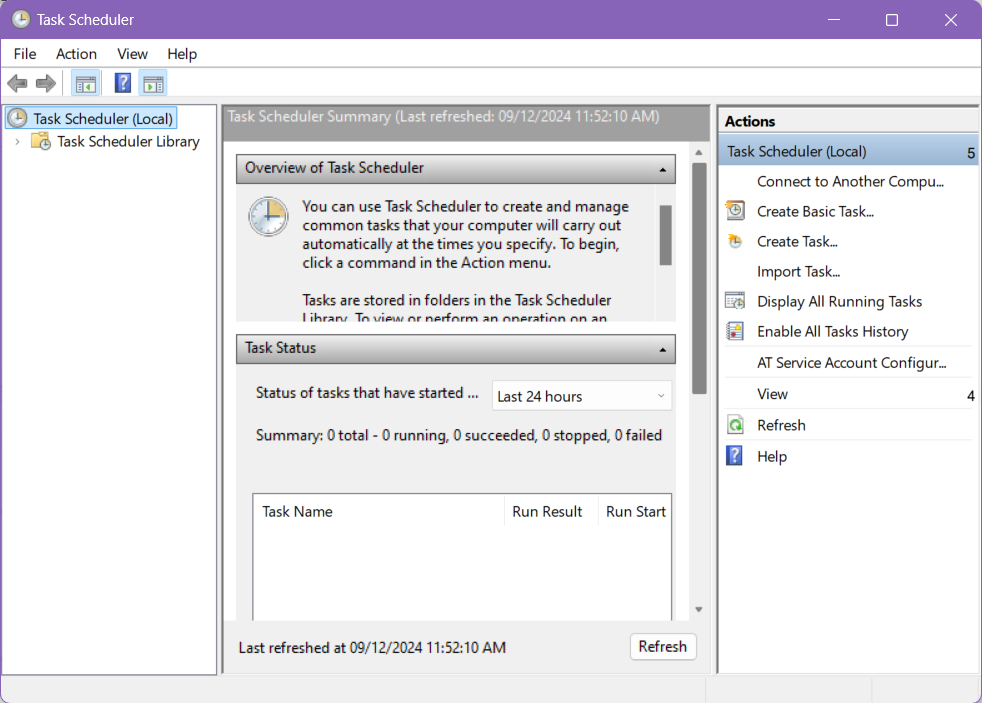
|  |
| --- |
| #1. Kết nối đến database Controls (Postgres)  # Thông tin kết nối trong file config.xml  #2. Sử dụng câu query để kiểm tra file config vừa chọn file config  SELECT \*  FROM file\_config  #3. Chạy file load\_to\_dw.py với 3 tham số <id\_config> <địa chỉ cụ thể file config.xml> <Ngày> (Optional, mặc định là ngày hôm nay, định dạng YYYY-MM-DD)  python load\_to\_dw.py 1 E:\Documents\NLU\DataWarehouse\project\config.xml |

## Scheduling

Sử dụng Task Scheduler trên Window

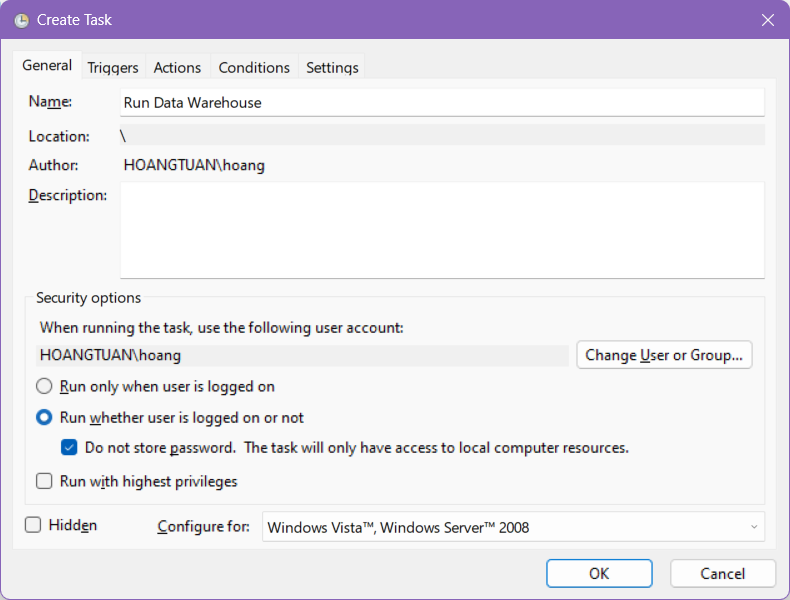
* Mở Task Scheduler: Nhấn **Win + S**, tìm kiếm Task Scheduler, và mở ứng dụng.
* Tạo task mới:

**1. Trong Task Scheduler, nhấn Create Task.**

****

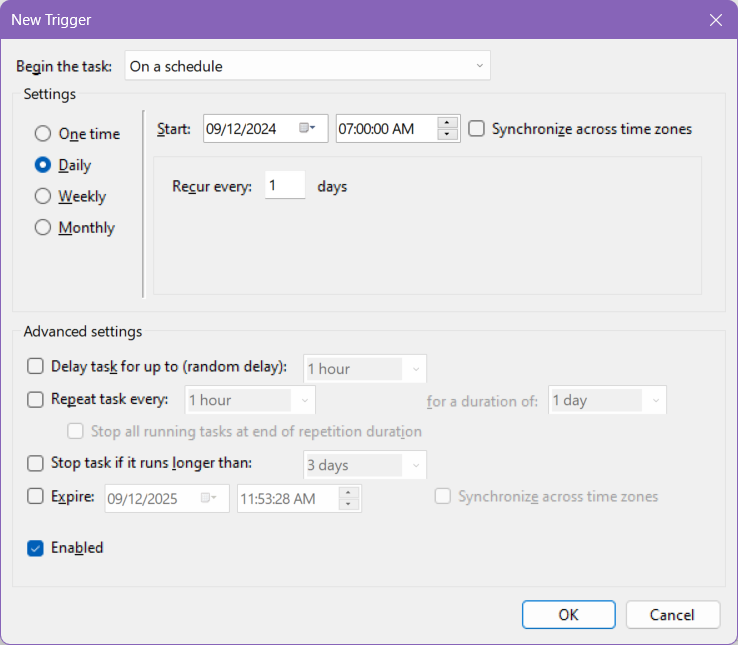
**2. Tab General:**

* Nhập tên task, ví dụ: Run Data Warehouse.
* Chọn **Run whether user is logged on or not**.
* Tick **Do not store password** (tùy chọn nếu không cần truy cập mạng).



**3.** **Tab Triggers**:

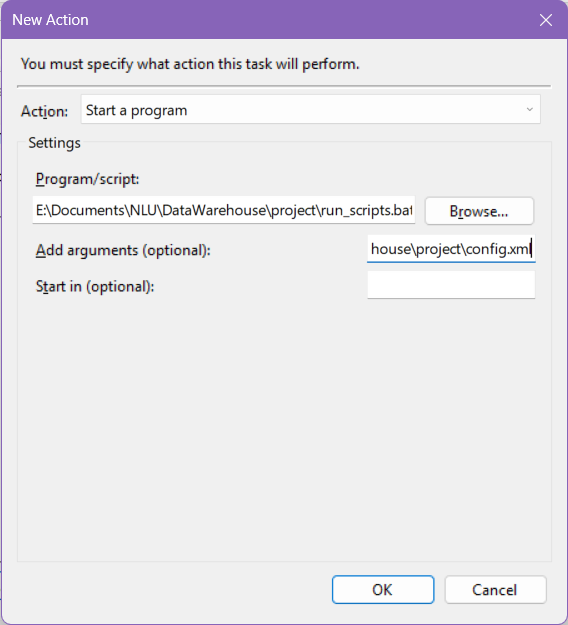
* Nhấn **New** để thêm trigger.
* Chọn loại lịch (ví dụ: hàng ngày, hàng giờ, hoặc giờ cụ thể).



**4**. **Tab Actions**:

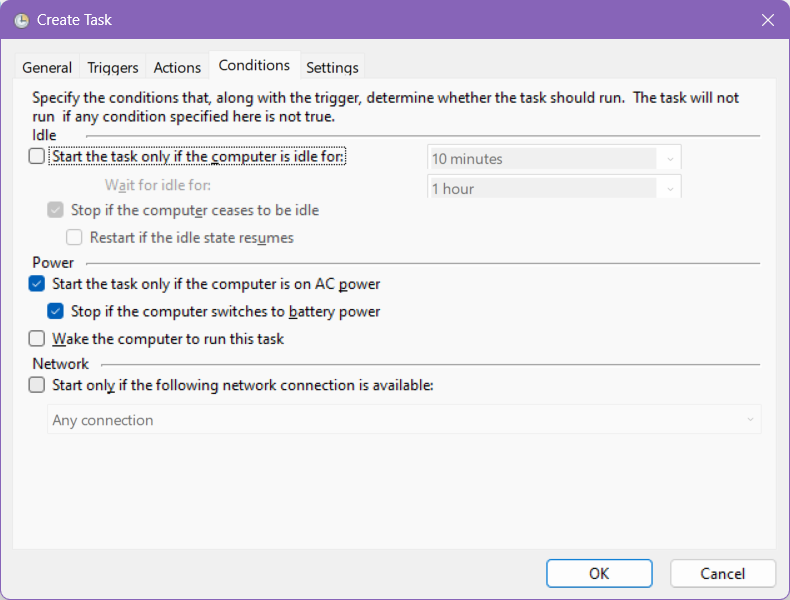
* Nhấn **New**.
* **Action:** Chọn **Start a Program**.
* **Program/script:** Chọn đường dẫn tới python.exe, ví dụ: E:\Documents\NLU\DataWarehouse\project\extract\_file.py
* **Add arguments (optional):** Nhập đường dẫn file script và các tham số id\_config, đường dẫn chính xác file config.xml, ví dụ:

1 E:\Documents\NLU\DataWarehouse\project\config.xml



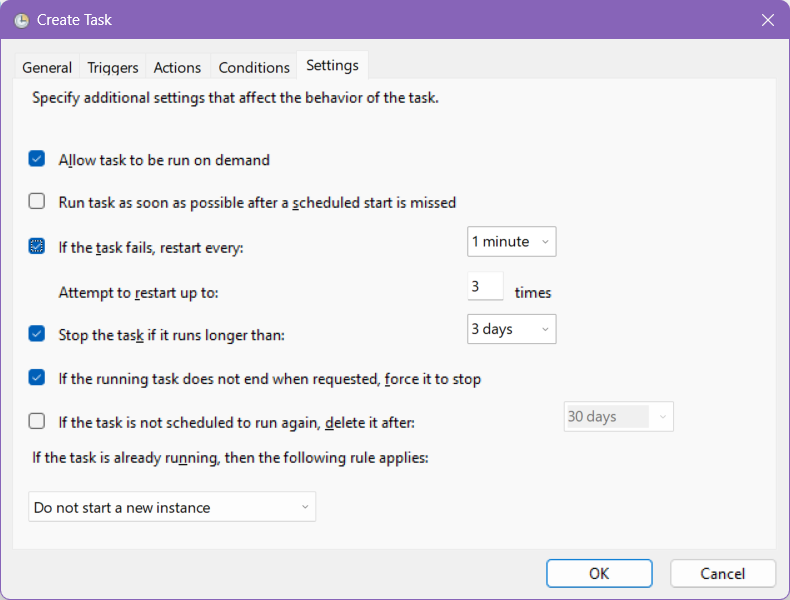
**5.Tab Conditions:**

* Bỏ tick **Start the task only if the computer is on AC power** nếu không muốn điều kiện này.



**6.Tab Settings**:

* Tick **Allow task to be run on demand**.
* Tick **If the task fails, restart every...** để tự động khởi động lại nếu xảy ra lỗi.

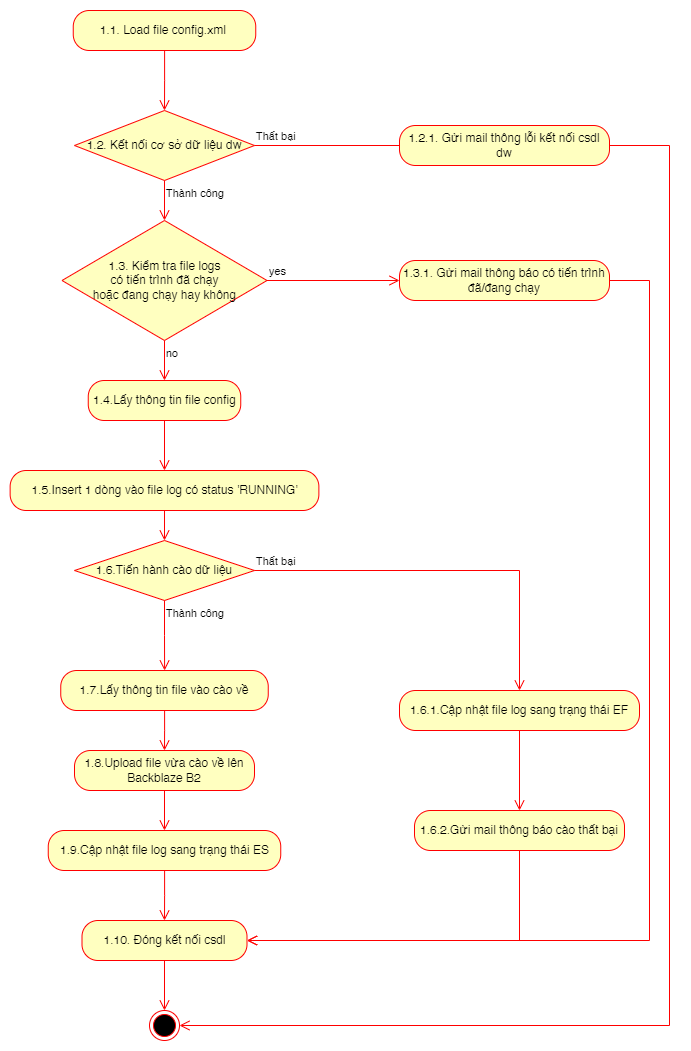


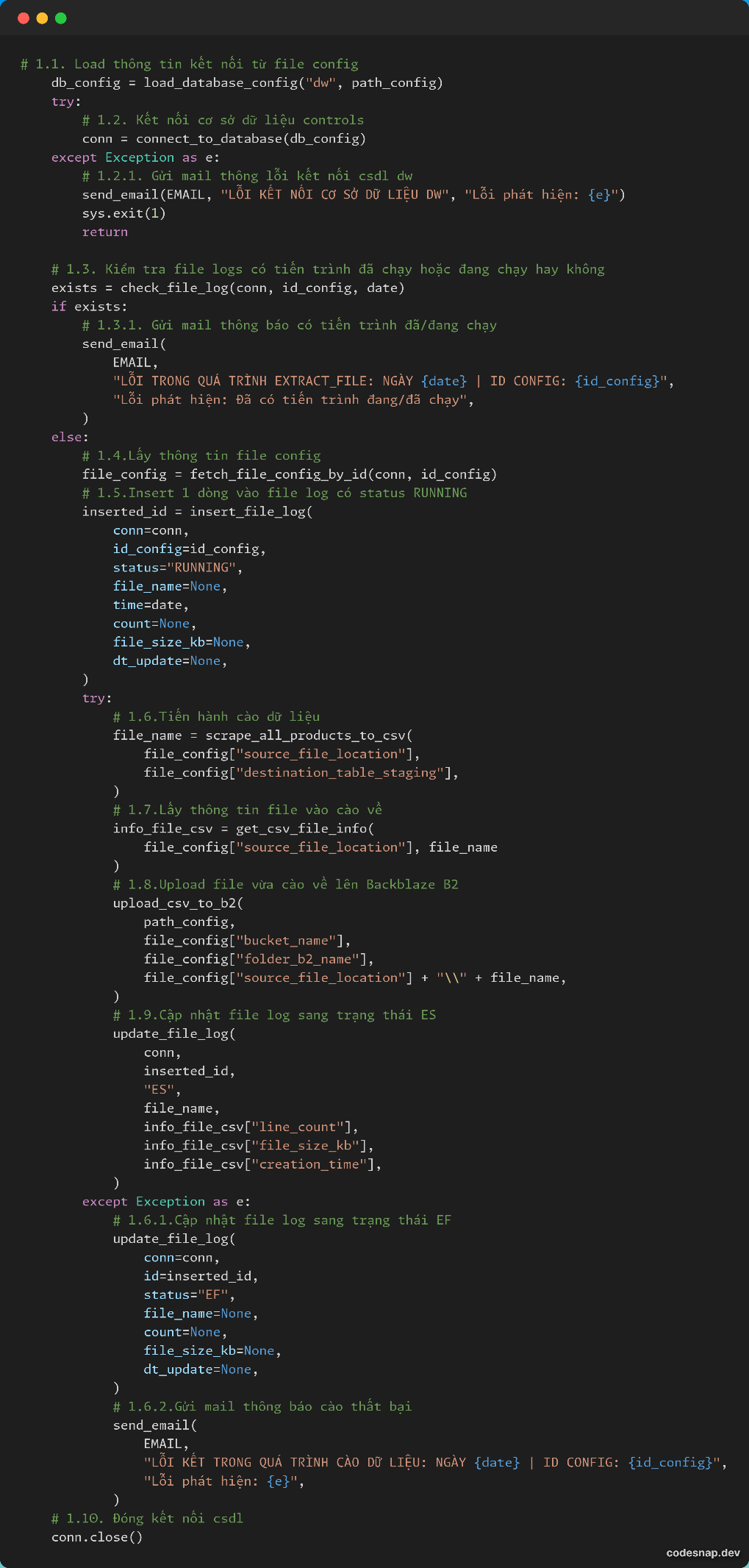
|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Subject** | **Schedule** | **Note** |
| 1 | Chạy với run\_scripts.bat | 7AM | Khi chạy sẽ tự động chạy các process theo thứ tự P1 (Extract file) 🡪 P2 (Load to staging) 🡪 P3 (Load to dw) 🡪 P4 (Load to dm). |

*\*Có thể tùy chỉnh chạy từng giai đoạn riêng lẻ*

## Logical Workflow

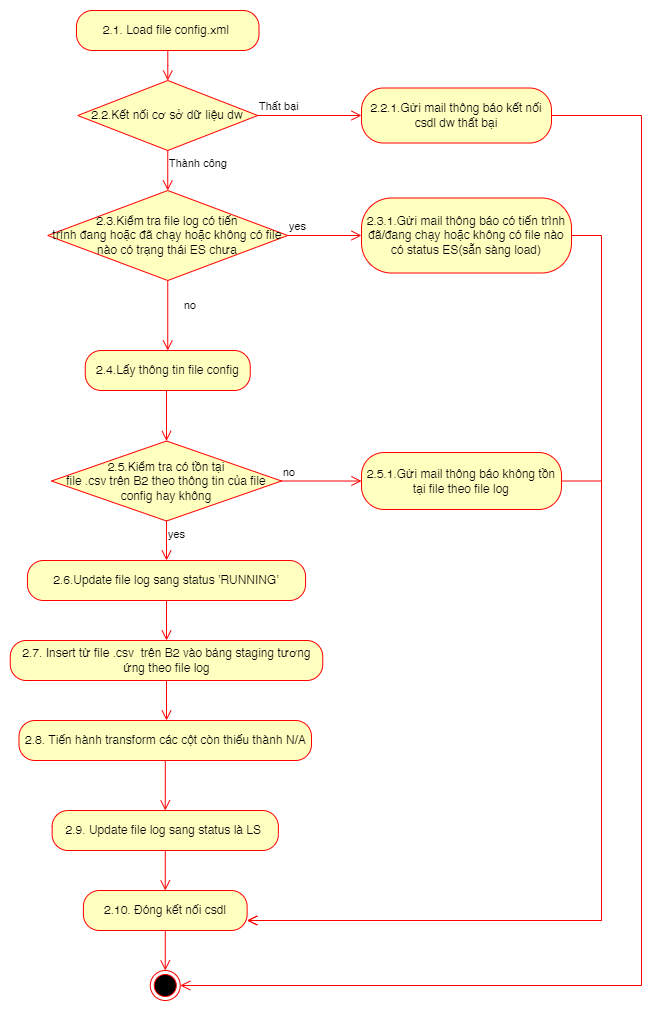
### 6.1. Extract file process





|  |
| --- |
| * 1. **load\_database\_config** |
| **Refer:**  **Purpose:** Lấy thông tin để kết nối database từ file config.xml đường dẫn chính xác file là input nhập vào.  **Input:** db\_name (tên database), path\_config (đường dẫn chính xác của file config.xml).  **Output:** 1 dictionary chứa thông tin kết nối database tương ứng. |
| * 1. **connect\_to\_database** |
| **Refer:**  **Purpose:** Kết nối với database Postgres.  **Input:** db\_config(1 dictionary chứa các thông tin kết nối).  **Output:** 1 biến Connection. |
| * 1. **check\_file\_log** |
| **Refer:**  **Purpose:** Kiểm tra file logs có tiến trình đã chạy hoặc đang chạy hay không.  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), date(ngày định dạng YYYY-MM-DD).  **Output:** True/False. |
| * 1. **fetch\_file\_config\_by\_id** |
| **Refer:**  **Purpose:** Lấy thông tin file config.  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config).  **Output:** 1 dictionary chứa thông tin của file config. |
| * 1. **insert\_file\_log** |
| **Refer:**  **Purpose:** Insert 1 dòng vào file log có status RUNNING  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), status (“RUNNING”), file\_name (tên file), time, count, file\_size\_kb (dung lượng của file), dt\_update  **Output:** 1 dòng dữ liệu vào file\_log chứa những thông tin trên, inserted\_id (id của dòng vừa được chèn) |
| * 1. **scrape\_all\_products\_to\_csv** |
| **Refer:**  **Purpose:** Cào dữ liệu  **Input:** source\_file\_location (thư mục chứa dữ liệu), destination\_table\_staging (bảng staging)  **Output:** file\_name (tên file) |
| **1.6.1. update\_file\_log** |
| **Refer:**  **Purpose:** Cập nhật file log sang trạng thái “EF”  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), status (“EF”), file\_name (tên file), time, count, file\_size\_kb (dung lượng của file), dt\_update  **Output:** File log trong csdl được cập nhật status sang “EF” |
| * 1. **get\_csv\_file\_info** |
| **Refer:**  **Purpose:** Lấy thông tin file vào cào về  **Input:** source\_file\_location (thư mục chứa dữ liệu), file\_name  **Output:** |
| * 1. **upload\_csv\_to\_b2** |
| **Refer:**  **Purpose:** Upload file vừa cào về lên Blackblaze B2  **Input:** path\_config (đường dẫn chính xác của file config.xml), tên bucket trong Backblaze B2 mà file sẽ được upload vào, tên thư mục (folder) trong bucket trên Backblaze B2 để lưu trữ file CSV, đường dẫn đầy đủ của file CSV vừa được cào về (nằm trên máy local), cần được upload lên Backblaze B2  **Output:** Upload thành công file CSV |
| * 1. **upload\_file\_log** |
| **Refer:**  **Purpose:** Cập nhật file log sang trạng thái ES  **Input:** conn (1 connection kết nối csdl), inserted\_id (id của dòng vừa được chèn ở mục 1.5), “ES” (cập nhật status mới sang “ES”), số dòng trong file CSV, kích thước của file CSV.  **Output:** Cập nhật thành công bản ghi trong file\_log |
| * 1. **conn.close** |
| **Refer:**  **Purpose:** Đóng kết nối csdl  **Input:** conn (1 connection kết nối csdl)  **Output:** Kết nối csdl được đóng thành công |
| **1.2.1., 1.3.1, 1.6.2 send\_email** |
| **Refer:**  **Purpose:** Gửi email qua Gmail SMTP  **Input:** to\_email(Email người nhận (str)), subject(Tiêu đề email (str)), body(Nội dung email (str)).  **Output:** Gửi lỗi qua email |

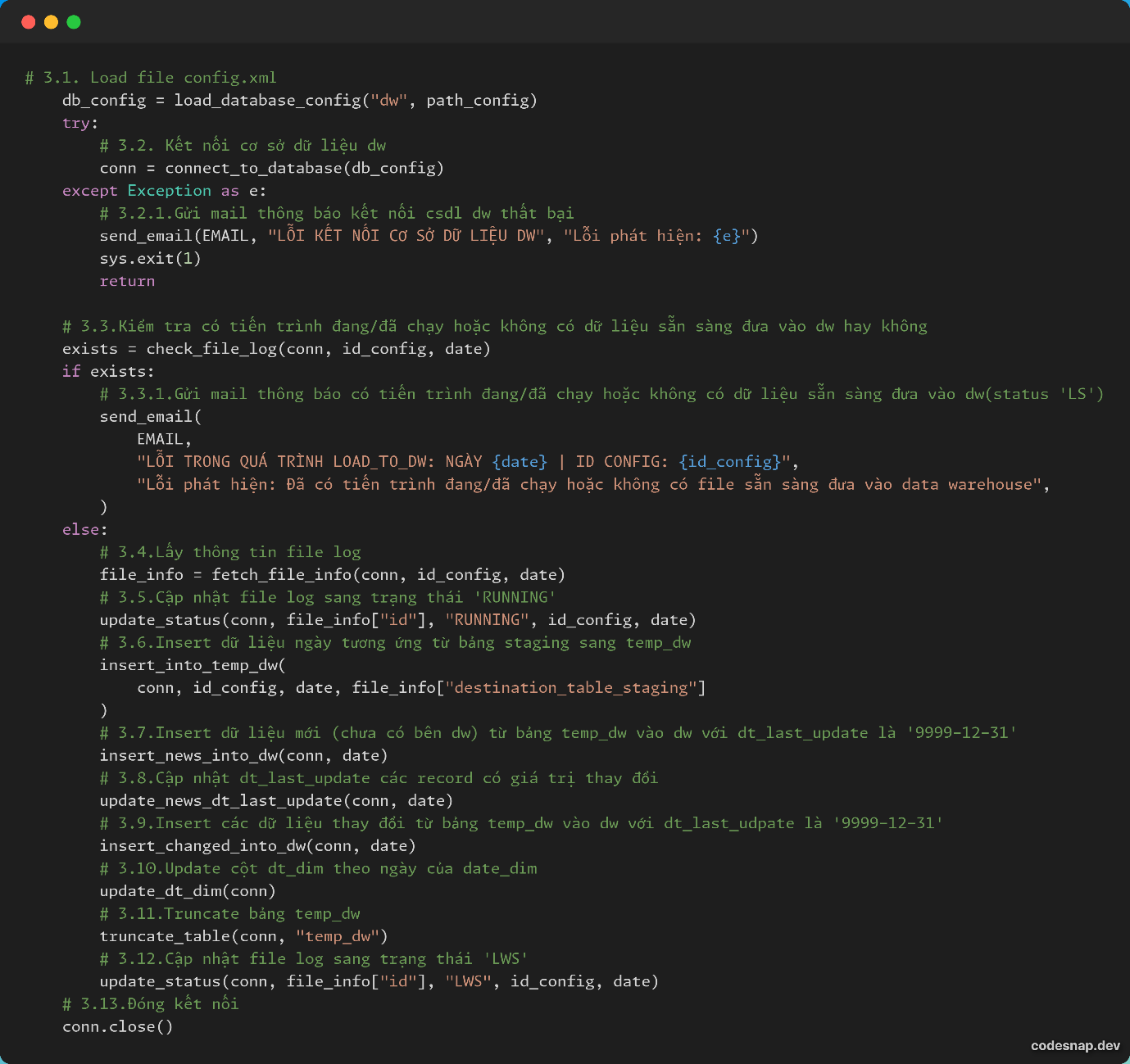
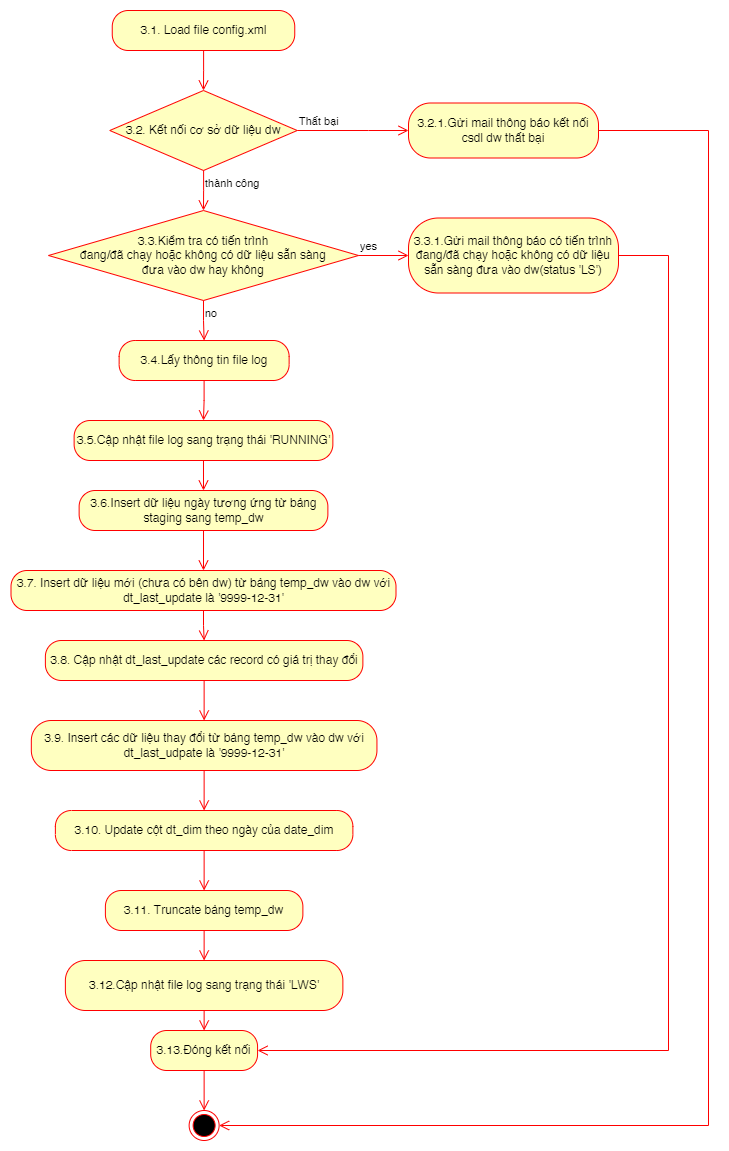
### 6.2. Load to staging process





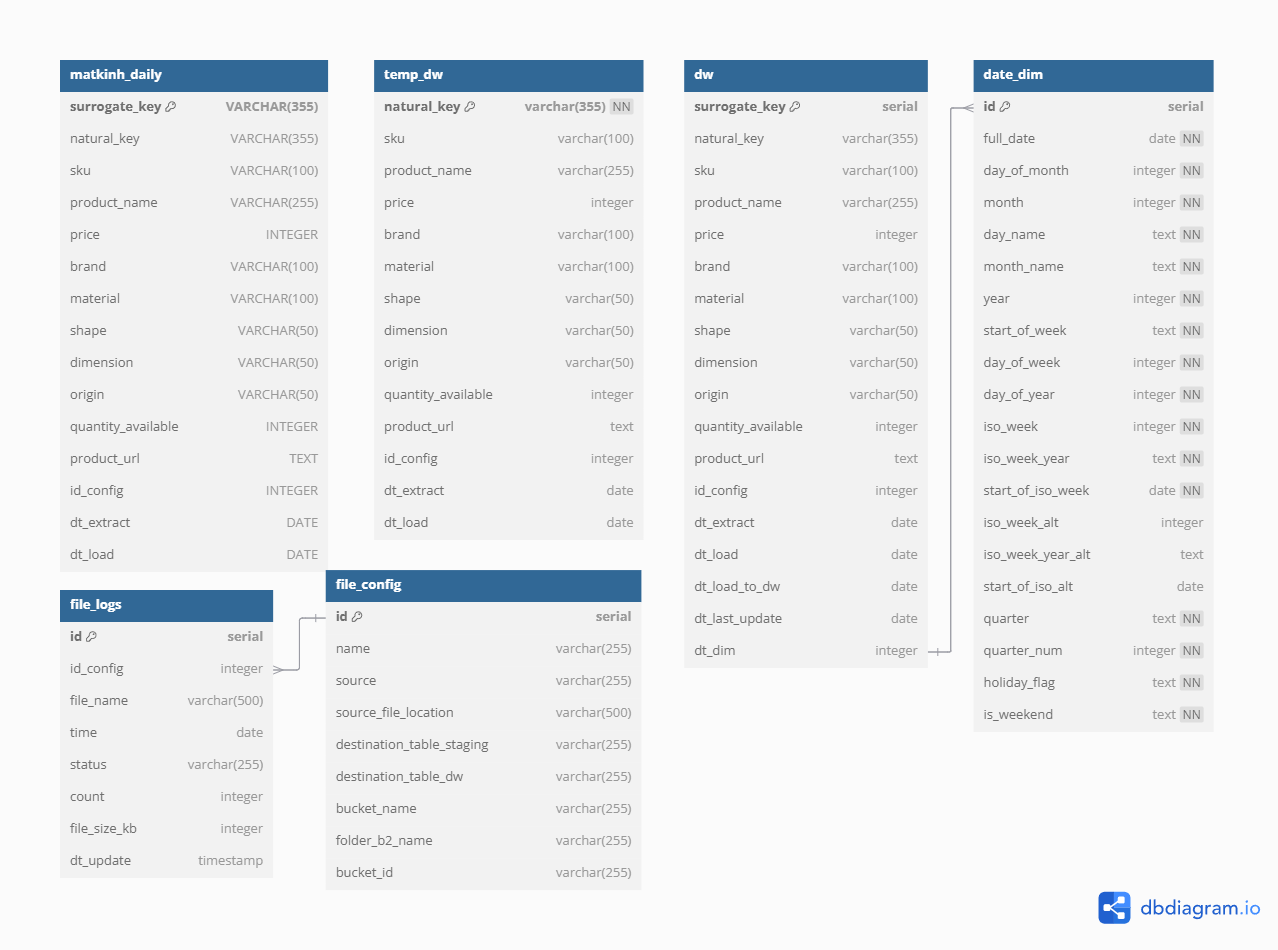
|  |
| --- |
| **2.1. load\_database\_config** |
| **Refer:**  **Purpose:** Lấy thông tin để kết nối database từ file config.xml đường dẫn chính xác file là input nhập vào.  **Input:** db\_name (tên database), path\_config (đường dẫn chính xác của file config.xml).  **Output:** 1 dictionary chứa thông tin kết nối database tương ứng. |
| **2.2. connect\_to\_database** |
| **Refer:**  **Purpose:** Kết nối với database Postgres.  **Input:** db\_config(1 dictionary chứa các thông tin kết nối).  **Output:** 1 biến Connection. |
| **2.3. check\_file\_log** |
| **Refer:**  **Purpose:** Kiểm tra file logs có tiến trình đã chạy hoặc đang chạy hay không.  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), date(ngày định dạng YYYY-MM-DD).  **Output:** True/False. |
| **2.4. fetch\_file\_info** |
| **Refer:**  **Purpose:** Lấy thông tin file\_config  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), date(ngày định dạng YYYY-MM-DD).  **Output:** In ra thông tin file |
| **2.5. check\_csv\_existed\_in\_b2** |
| **Refer:**  **Purpose:** Lấy thông tin file\_config  **Input:** path\_config (đường dẫn chính xác của file config.xml), Tên bucket trong Backblaze B2 (nơi file CSV được lưu trữ), Tên thư mục trong bucket, file\_name (tên file csv cần kiểm tra)  **Output:** True/False |
| **2.6. update\_status** |
| **Refer:**  **Purpose:** Update file\_log sang status “RUNNING”  **Input:** conn (1 connection kết nối csdl), id của file\_log cần được cập nhật, id\_config (id của file config), time, “RUNNING” (status cần được cập nhật)  **Output:** Cập nhật trạng thái file log sang “RUNNING” |
| **2.7. insert\_to\_table\_from\_b2** |
| **Refer:**  **Purpose:** Lấy thông tin file\_config  **Input:** path\_config (đường dẫn chính xác của file config.xml), ID bucket chứa file, Tên bucket chứa file, Tên thư mục chứa file, file\_name (tên file csv cần kiểm tra), destination\_table\_staging (tên bảng staging), id\_config, time, date  **Output:** Dữ liệu từ file csv được tải và chèn vào bảng staging trong csdl |
| **2.8. transform\_data** |
| **Refer:**  **Purpose:** Transform các cột còn thiếu thành N/A  **Input:**  conn (1 connection kết nối csdl), destination\_table\_staging (tên bảng staging)  **Output:** Các giá trị null được thay đổi thành giá trị N/A |
| **2.9. update\_status** |
| **Refer:**  **Purpose:** Update file\_log sang status “LS”  **Input:** conn (1 connection kết nối csdl), id của file\_log cần được cập nhật, id\_config (id của file config), time, “LS” (status cần được cập nhật)  **Output:** Cập nhật trạng thái file log sang “LS” |
| **2.10. conn.close** |
| **Refer:**  **Purpose:** Đóng kết nối csdl  **Input:** conn (1 connection kết nối csdl)  **Output:** Kết nối csdl được đóng thành công |
| **2.2.1, 2.3.1, 2.5.1. send\_email** |
| **Refer:**  **Purpose:** Gửi email qua Gmail SMTP  **Input:** to\_email(Email người nhận (str)), subject(Tiêu đề email (str)), body(Nội dung email (str)).  **Output:** Gửi lỗi qua email |

### 6.3. Load to dw process



|  |
| --- |
| **3.1. load\_database\_config** |
| **Refer:**  **Purpose:** Lấy thông tin để kết nối database từ file config.xml đường dẫn chính xác file là input nhập vào.  **Input:** db\_name (tên database), config\_path(đường dẫn chính xác của file config.xml).  **Output:** 1 dictionary chứa thông tin kết nối database tương ứng. |
| **3.2. connect\_to\_database** |
| **Refer:**  **Purpose:** Kết nối với database Postgres.  **Input:** db\_config(1 dictionary chứa các thông tin kết nối).  **Output:** 1 biến Connection. |
| **3.3. check\_file\_log** |
| **Refer:**  **Purpose:** Kiểm tra file logs có tiến trình đã chạy hoặc đang chạy hay không.  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config), date(ngày định dạng YYYY-MM-DD).  **Output:** True/False. |
| **3.4. fetch\_file\_config\_by\_id** |
| **Refer:**  **Purpose:** Lấy thông tin file config.  **Input:** conn (1 connection kết nối csdl), id\_config (id của file config).  **Output:** 1 dictionary chứa thông tin của file config. |
| **3.5. update\_status** |
| **Refer:**  **Purpose:** Cập nhật cột ‘status’ và ‘time’record với điều kiện trong bảng file\_logs.  **Input:** conn(1 connection kết nối csdl), record\_id(id của file logs), status(trạng thái muốn cập nhật), id\_config(id của file config), time\_value(ngày định dạng YYYY-MM-DD).  **Output:** |
| **3.6. insert\_into\_temp\_dw** |
| **Refer:**  **Purpose:** chèn dữ liệu từ bảng staging vào bảng temp\_dw với điều kiện `id\_config` và `dt\_load`.  **Input:** conn(Kết nối PostgreSQL), id\_config(giá trị id\_config cần kiểm tra), date(Ngày `dt\_load` cần kiểm tra (dạng chuỗi 'YYYY-MM-DD')), table\_staging(Tên bảng staging để lấy dữ liệu nguồn).  **Output:** True/False |
| **3.7. insert\_news\_into\_dw** |
| **Refer:**  **Purpose:** Insert dữ liệu mới (chưa có bên dw) từ bảng temp\_dw vào dw với dt\_last\_update là '9999-12-31'  **Input:** conn(Kết nối PostgreSQL), dt\_load\_to\_dw( ngày (dạng chuỗi 'YYYY-MM-DD')).  **Output:** |
| **3.8. update\_news\_dt\_last\_update** |
| **Refer:**  **Purpose:** Cập nhật dt\_last\_update các record có giá trị thay đổi  **Input:** conn(Kết nối PostgreSQL), dt\_load\_to\_dw( ngày (dạng chuỗi 'YYYY-MM-DD')).  **Output:** |
| **3.9 insert\_changed\_into\_dw** |
| **Refer:**  **Purpose:** Insert các dữ liệu thay đổi từ bảng temp\_dw vào dw với dt\_last\_udpate là '9999-12-31'  **Input:** conn(Kết nối PostgreSQL), dt\_load\_to\_dw( ngày (dạng chuỗi 'YYYY-MM-DD')).  **Output:** |
| **3.10. update\_dt\_dim** |
| **Refer:**  **Purpose:** Update cột dt\_dim theo ngày của date\_dim  **Input:** conn(Kết nối PostgreSQL).  **Output:** |
| **3.11. truncate\_table** |
| **Refer:**  **Purpose:** Truncate bảng có tên là input nhập vào  **Input:** conn(Kết nối PostgreSQL), table\_name(tên bảng cần truncate).  **Output:** |
| **3.3.1, 3.2.1. send\_mail** |
| **Refer:**  **Purpose:** Gửi email qua Gmail SMTP  **Input:** to\_email(Email người nhận (str)), subject(Tiêu đề email (str)), body(Nội dung email (str)).  **Output:** |

## Database Structure



*(Database digram trong dw)*

* Bảng matkinh\_daily: chứa thông tin dữ liệu từ file .csv cào về hằng ngày:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | surrogate\_key | serial | 1406 |
| 2 | natural\_key | varchar(355) | BOLON BT6010-BT6010 B12 |
| 3 | sku | varchar(100) | BT6010 B12 |
| 4 | product\_name | varchar(255) | BOLON BT6010 |
| 5 | price | integer | 3980000 |
| 6 | brand | varchar(100) | BOLON |
| 7 | material | varchar(100) | Nhựa/ Titanium |
| 8 | shape | varchar(50) | Vuông |
| 9 | dimension | varchar(50) | 49-19-150 |
| 10 | origin | varchar(50) | Trung |
| 11 | quantity\_available | integer | 20 |
| 12 | product\_url | text | https://kinhmatviettin.vn/products/bolon-bt6010-b12 |
| 13 | id\_config | integer | 1 |
| 14 | dt\_extract | date | 2024-12-08 |
| 15 | dt\_load | date | 2024-12-08 |

* Bảng dw: bảng chứa dữ liệu dw:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | natural\_key | varchar(355) | BOLON BT6010-BT6010 B12 |
| 2 | sku | varchar(100) | BT6010 B12 |
| 3 | product\_name | varchar(255) | BOLON BT6010 |
| 4 | price | integer | 3980000 |
| 5 | brand | varchar(100) | BOLON |
| 6 | material | varchar(100) | Nhựa/ Titanium |
| 7 | shape | varchar(50) | Vuông |
| 8 | dimension | varchar(50) | 49-19-150 |
| 9 | origin | varchar(50) | Trung |
| 10 | quantity\_available | integer | 20 |
| 11 | product\_url | text | https://kinhmatviettin.vn/products/bolon-bt6010-b12 |
| 12 | id\_config | integer | 1 |
| 13 | dt\_extract | date | 2024-12-08 |
| 14 | dt\_load | date | 2024-12-08 |

* Bảng matkinh\_daily: chứa thông tin dữ liệu từ file .csv cào về hằng ngày.

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | surrogate\_key | serial | 1406 |
| 2 | natural\_key | varchar(355) | BOLON BT6010-BT6010 B12 |
| 3 | sku | varchar(100) | BT6010 B12 |
| 4 | product\_name | varchar(255) | BOLON BT6010 |
| 5 | price | integer | 3980000 |
| 6 | brand | varchar(100) | BOLON |
| 7 | material | varchar(100) | Nhựa/ Titanium |
| 8 | shape | varchar(50) | Vuông |
| 9 | dimension | varchar(50) | 49-19-150 |
| 10 | origin | varchar(50) | Trung |
| 11 | quantity\_available | integer | 20 |
| 12 | product\_url | text | https://kinhmatviettin.vn/products/bolon-bt6010-b12 |
| 13 | id\_config | integer | 1 |
| 14 | dt\_extract | date | 2024-12-08 |
| 15 | dt\_load | date | 2024-12-08 |
| 16 | dt\_load\_dw | date | 2024-12-08 |
| 17 | dt\_load\_last\_update | date | 9999-12-31 |
| 18 | dt\_dim | integer | 7281 |

* Bảng date\_dim:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | id | serial | 1 |
| 2 | full\_date | date | 2005-01-01 |
| 3 | day\_of\_month | integer | 1 |
| 4 | month | integer | 1 |
| 5 | day\_name | text | Saturday |
| 6 | month\_name | text | January |
| 7 | year | integer | 2005 |
| 8 | start\_of\_week | text | 2005-Jan |
| 9 | day\_of\_week | integer | 1 |
| 10 | day\_of\_year | integer | 1 |
| 11 | iso\_week | integer | 52 |
| 12 | iso\_week\_year | text | 2004-W52 |
| 13 | start\_of\_iso\_week | date | 2004-12-26 |
| 14 | iso\_week\_alt | integer | 53 |
| 15 | iso\_week\_year\_alt | text | 2004-W53 |
| 16 | start\_of\_iso\_alt | date | 2004-12-27 |
| 17 | quarter | text | 2005-Q01 |
| 18 | quarter\_num | integer | 1 |
| 19 | holiday\_flag | text | Non-Holiday |
| 20 | is\_weekend | text | Weekend |

* Bảng file\_config: chứa các thông tin tùy chỉnh để chạy chương trình:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | id | serial | 1 |
| 2 | name | varchar(255) | Mắt kính Việt Tín |
| 3 | source | varchar(255) | E:\Documents\NLU\DataWarehouse\project\daily |
| 4 | source\_file\_location | varchar(500) | matkinh\_daily |
| 5 | destination\_table\_staging | varchar(255) | matkinh\_daily |
| 6 | destination\_table\_dw | varchar(255) | dw |
| 7 | bucket\_name | varchar(255) | dw-nlu-storage |
| 8 | folder\_b2\_name | varchar(255) | daily |
| 9 | bucket\_id | varchar(255) | a5a379fab4009ce2983e0815 |

* Bảng file\_logs: chứa dữ liệu ghi lại lịch sử làm việc của dự án:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | id | serial | 1 |
| 2 | id\_config | integer | 1 |
| 3 | file\_name | varchar(500) | data\_matkinh\_daily\_2024-12-07\_kinhmatviettin.vn.csv |
| 4 | time | date | 2024-12-07 |
| 5 | status | varchar(255) | LWS |
| 6 | count | integer | 100 |
| 7 | file\_size\_kb | integer | 16 |
| 8 | dt\_update | timestamp | 2024-12-06 17:37:42.000000 |

## Data feed specification

* File format: Các định dạng tệp hiện được chấp nhận là ASCII CSV (các giá trị được phân tách bằng dấu phẩy).
* Data size: < 1mb/ngày
* Data location: [https://f005.backblazeb2.com/file/dw-nlu-storage/…](https://f005.backblazeb2.com/file/dw-nlu-storage/%E2%80%A6)
* Name convention: data{id config}\_{table\_staging\_destiantion}\_YYYY-MM-DD\_{source}.csv

YYY-MM-DD là ngày cào dữ liệu.

source là nguồn lấy dữ liệu.

table\_staging\_destination là tên bảng khi đưa vào staging.

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field name | Format | Description(Sample, value) |
| 1 | surrogate\_key | serial | 1406 |
| 2 | natural\_key | varchar(355) | BOLON BT6010-BT6010 B12 |
| 3 | sku | varchar(100) | BT6010 B12 |
| 4 | product\_name | varchar(255) | BOLON BT6010 |
| 5 | price | integer | 3980000 |
| 6 | brand | varchar(100) | BOLON |
| 7 | material | varchar(100) | Nhựa/ Titanium |
| 8 | shape | varchar(50) | Vuông |
| 9 | dimension | varchar(50) | 49-19-150 |
| 10 | origin | varchar(50) | Trung |
| 11 | quantity\_available | integer | 20 |
| 12 | product\_url | text | https://kinhmatviettin.vn/products/bolon-bt6010-b12 |
| 13 | id\_config | integer | 1 |
| 14 | dt\_extract | date | 2024-12-08 |
| 15 | dt\_load | date | 2024-12-08 |
| 16 | dt\_load\_dw | date | 2024-12-08 |
| 17 | dt\_load\_last\_update | date | 9999-12-31 |
| 18 | dt\_dim | integer | 7281 |

## Script setup tables in Postgres Database

* Bảng matkinh\_daily

|  |
| --- |
| create table matkinh\_daily  (  surrogate\_key serial  primary key,  natural\_key varchar(355),  sku varchar(100),  product\_name varchar(255),  price integer,  brand varchar(100),  material varchar(100),  shape varchar(50),  dimension varchar(50),  origin varchar(50),  quantity\_available integer,  product\_url text,  id\_config integer,  dt\_extract date,  dt\_load date  ); |

* Bảng dw

|  |
| --- |
| create table dw  (  surrogate\_key serial  primary key,  natural\_key varchar(355),  sku varchar(100),  product\_name varchar(255),  price integer,  brand varchar(100),  material varchar(100),  shape varchar(50),  dimension varchar(50),  origin varchar(50),  quantity\_available integer,  product\_url text,  id\_config integer,  dt\_extract date,  dt\_load date,  dt\_load\_to\_dw date,  dt\_last\_update date,  dt\_dim integer  ); |

* Bảng file\_config

|  |
| --- |
| create table file\_config  (  id serial  primary key,  name varchar(255),  source varchar(255),  source\_file\_location varchar(500),  destination\_table\_staging varchar(255),  destination\_table\_dw varchar(255),  bucket\_name varchar(255),  folder\_b2\_name varchar(255),  bucket\_id varchar(255)  ); |

* Bảng file\_logs

|  |
| --- |
| create table file\_logs  (  id serial  primary key,  id\_config integer  references file\_config  on delete cascade,  file\_name varchar(500),  time date,  status varchar(255),  count integer,  file\_size\_kb integer,  dt\_update timestamp  ); |

* Bảng temp\_dw

|  |
| --- |
| create table temp\_dw  (  natural\_key varchar(355) not null  primary key,  sku varchar(100),  product\_name varchar(255),  price integer,  brand varchar(100),  material varchar(100),  shape varchar(50),  dimension varchar(50),  origin varchar(50),  quantity\_available integer,  product\_url text,  id\_config integer,  dt\_extract date,  dt\_load date  ); |

* Bảng date\_dim

|  |
| --- |
| create table date\_dim  (  id serial  primary key,  full\_date date not null,  day\_of\_month integer not null,  month integer not null,  day\_name text not null,  month\_name text not null,  year integer not null,  start\_of\_week text not null,  day\_of\_week integer not null,  day\_of\_year integer not null,  iso\_week integer not null,  iso\_week\_year text not null,  start\_of\_iso\_week date not null,  iso\_week\_alt integer,  iso\_week\_year\_alt text,  start\_of\_iso\_alt date,  quarter text not null,  quarter\_num integer not null,  holiday\_flag text not null,  is\_weekend text not null  ); |

Chạy file insert\_date\_dim.py để insert dữ liệu từ file date\_dim.csv vào bảng date\_dim:

Giả sử đường dẫn là “E:\Documents\NLU\DataWarehouse\project”

Yêu cầu máy có python và kết nối Internet

Mở cmd tại folder chứa các file cần thiết chạy lệnh sau:

|  |
| --- |
| # Tham số là đường dẫn chính xác file config.xml và date\_dim.csv  python insert\_date\_dim.py E:\Documents\NLU\DataWarehouse\project\config.xml E:\Documents\NLU\DataWarehouse\project\date\_dim.csv |