REPORT 2: DATA PIPELINE ARCHITECTURE & ETL PROCESS

- **Môn học:** ADY201m: AI, DATA SCIENCE WITH PYTHON & SQL
- **Dự án:** Sentiment Analysis for Foody Reviews
- **Giai đoạn:** Tuần 3-4

---

1. TỔNG QUAN DỰ ÁN (PROJECT OVERVIEW)

1.1. Mục tiêu

Xây dựng một hệ thống Data Pipeline tự động (End-to-End) nhằm thu thập, lưu trữ tập trung và phân tích dữ liệu đánh giá ẩm thực từ nền tảng Foody.vn. Hệ thống phục vụ các mục đích chính:

- **Data Storage:** Lưu trữ dữ liệu thô (Raw Data) an toàn trên Data Lake.
- **Data Processing:** Làm sạch và chuẩn hóa dữ liệu đa vùng miền.
- **Analytics:** Phân tích cảm xúc khách hàng (Sentiment Analysis) và xu hướng tiêu dùng.

1.2. Phạm vi dữ liệu (Data Scope)

- **Nguồn dữ liệu:** Foody.vn
- **Phạm vi địa lý:** * Hà Nội (Đại diện Miền Bắc)
  - Bình Định (Đại diện Miền Trung)
  - TP.HCM (Đại diện Miền Nam)
- **Dung lượng xử lý:** ~14,773 bản ghi sạch (Cleaned Rows) sau khi qua xử lý.

---

2. KIẾN TRÚC HỆ THỐNG (DATA PIPELINE ARCHITECTURE)

Hệ thống được thiết kế theo mô hình **ELT/ETL** hiện đại, triển khai hoàn toàn trên **Docker Containers**, đảm bảo tính tách biệt giữa các tầng dữ liệu.

Sơ đồ luồng dữ liệu (Data Flow)

1. **Ingestion Layer (Thu thập):**
   - Sử dụng Python (`Selenium`, `Requests`) để crawl dữ liệu từ web.
   - Phân luồng dữ liệu theo khu vực địa lý ngay từ đầu vào.
2. **Data Lake Layer (Lưu trữ thô - Raw Zone):**
   - **Công nghệ:** MinIO (S3 Compatible Object Storage).
   - **Định dạng:** JSON/CSV (Giữ nguyên cấu trúc gốc).
3. **Processing Layer (Xử lý & Làm sạch):**

- **Công nghệ:** Python (Pandas, Numpy, NLP Libraries).
- **Nhiệm vụ:** Đọc dữ liệu từ MinIO -> Transform -> Chuẩn bị nạp DB.

4. **Data Warehouse Layer (Kho dữ liệu - Clean Zone):**
   - **Công nghệ:** PostgreSQL.
   - **Schema:** Star Schema (tối ưu cho truy vấn phân tích).

---

3. QUY TRÌNH ETL (EXTRACT - TRANSFORM - LOAD)

Đây là thành phần cốt lõi ("Heavy Workload") của Report 2, xử lý việc chuyển đổi dữ liệu thô thành dữ liệu có giá trị (Insights).

3.1. Extract (Trích xuất)

- Kết nối trực tiếp tới MinIO Server nội bộ.
- Trích xuất dữ liệu từ các phân vùng (partitions) file: `reviews_MienBac`, `reviews_MienTrung`, `reviews_MienNam`.
- Ghi nhận nguồn gốc file vào metadata (`region`) để phục vụ Data Lineage (Truy xuất nguồn gốc).

3.2. Transform (Chuyển đổi & Làm sạch)

Quy trình xử lý dữ liệu trải qua các bước nghiêm ngặt:

- **Data Cleaning (Làm sạch văn bản):**
  - Loại bỏ nhiễu: HTML tags, icon, emoji, ký tự đặc biệt không có ý nghĩa.
  - Xử lý văn bản: Lowercase toàn bộ comment để phục vụ mô hình NLP.
  - Output: Cột `comment_clean`.
- **Normalization (Chuẩn hóa địa danh):**
  - Áp dụng kỹ thuật **Slugify** để chuẩn hóa tên thành phố.
  - Quy tắc: Chuyển tiếng Việt có dấu thành không dấu, nối bằng gạch ngang.
  - *Ví dụ:* `Hà Nội` -> `ha-noi`, `Bình Định` -> `binh-dinh`.
  - *Mục đích:* Tránh lỗi Encoding vùng miền và tối ưu hóa tốc độ Indexing trong Database.
- **Deduplication (Lọc trùng):**
  - Kiểm tra trùng lặp dựa trên khóa chính (`review_id`) và các trường định danh phụ.
  - **Kết quả:** Loại bỏ dữ liệu trùng lặp, đảm bảo tính duy nhất (Uniqueness) cho 14,773 bản ghi.
- **Feature Engineering (Tạo đặc trưng):**

– Gán nhãn `sentiment_label`: Tự động phân loại **Positive/Negative/Neutral** dựa trên rating và từ khóa.

3.3. Load (Nạp dữ liệu)

- Thực hiện nạp dữ liệu vào bảng `fact_reviews` trong PostgreSQL.
- Sử dụng cơ chế **Upsert** hoặc **Batch Insert** để đảm bảo hiệu năng cao.

---

4. CẤU TRÚC DỮ LIỆU CUỐI CÙNG (DATA SCHEMA)

Sau quá trình ETL, dữ liệu trong Database `foody_analytics` đạt chuẩn cấu trúc (Structured Data):

Tên trường	Kiểu dữ liệu	Mô tả	Ghi chú
review_id	`VARCHAR`	Mã định danh review	*Primary Key*
username	`VARCHAR`	Tên người dùng	*Dùng để lọc trùng*
restaurant_name	`VARCHAR`	Tên nhà hàng/quán ăn	
city	`VARCHAR`	Thành phố (Slug)	*Indexed (ha-noi, binh-dinh)*
region	`VARCHAR`	Nguồn gốc file	*MinIO Path*
rating	`FLOAT`	Điểm đánh giá	*Scale: 1.0 - 10.0*
sentiment_label	`VARCHAR`	Nhãn cảm xúc	*Positive/Negative/Neutral*
comment_clean	`TEXT`	Nội dung đã làm sạch	*NLP ready*

5. KẾT LUẬN & ĐÁNH GIÁ (CONCLUSION)

- **Mức độ hoàn thành:** Đã xây dựng thành công luồng dữ liệu tự động từ Web -> Data Lake (MinIO) -> Data Warehouse (PostgreSQL).
- **Chất lượng dữ liệu:** * Dữ liệu sạch, không trùng lặp.
  - Đã giải quyết được bài toán đồng bộ format vùng miền (Slugify).
  - Sẵn sàng cho các truy vấn SQL phức tạp kiểm tra giả thuyết.
- **Tính mở rộng:** Hệ thống chạy trên Docker, dễ dàng mở rộng thêm các container xử lý hoặc Visualization (như Metabase/PowerBI) trong giai đoạn tiếp theo.