BỘ THÔNG TIN VÀ TRUYỀN THÔNG

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÁO CÁO THỰC TẬP CƠ SỞ

ĐỀ TÀI

Triển khai Big Data

Giảng viên hướng dẫn : Kim Ngọc Bách

Họ và tên : Ngô Vũ Minh Quý

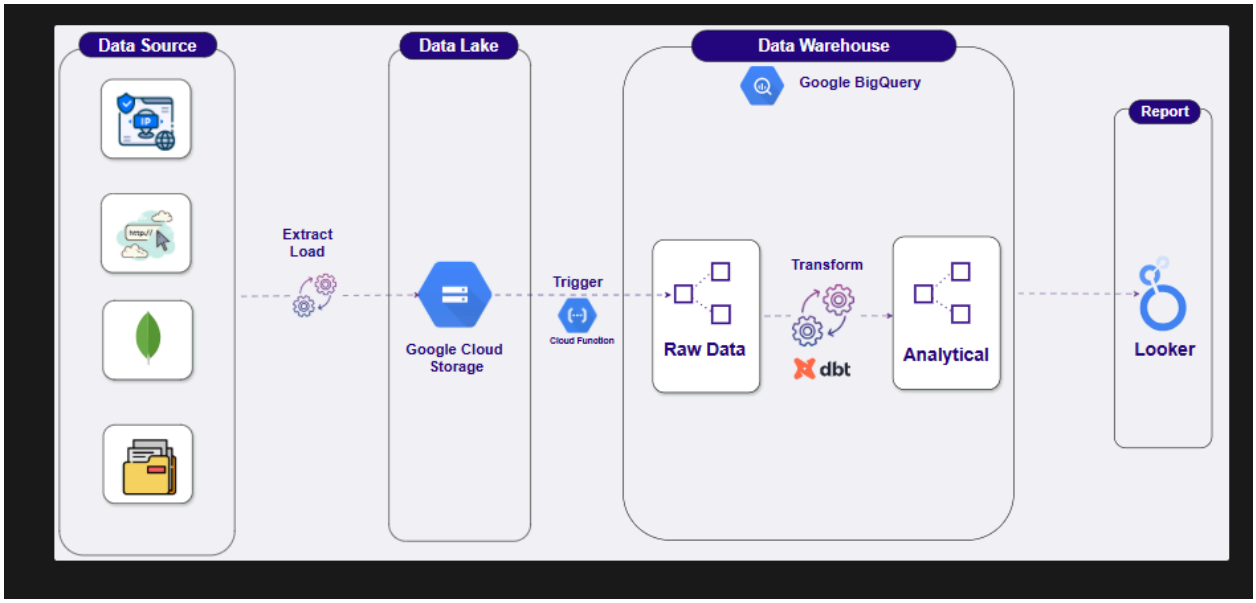
Mã sinh viên : B22DCVT427

Lớp : E22CQCN02-B

MỤC LỤC

I. Tổng quan về đề tài

→ Lý do chọn đề tài



-Trong thời đại công nghệ số, dữ liệu đang trở thành tài sản quan trọng bậc nhất đối với doanh nghiệp và tổ chức. Việc thu thập, xử lý và phân tích dữ liệu lớn (Big Data) không chỉ giúp cải thiện hiệu suất vận hành mà còn mang đến các insight giá trị hỗ trợ ra quyết định chiến lược. Tuy nhiên, để khai thác hiệu quả Big Data, cần có một hạ tầng dữ liệu hiện đại, có khả năng mở rộng, tự động hóa và dễ dàng trực quan hóa.

-Với sự hỗ trợ từ các nền tảng như Google Cloud Platform (GCP) và công cụ mã nguồn mở, đề tài "Triển khai Big Data" là cơ hội để sinh viên làm quen với các công nghệ thực tế đang được doanh nghiệp sử dụng, từ đó xây dựng kỹ năng nền tảng vững chắc trong lĩnh vực Data Engineering và Analytics.

→ Mục tiêu đề tài

Xây dựng một hệ thống data pipeline hoàn chỉnh, từ thu thập dữ liệu thô đến xử lý, lưu trữ, biến đổi và trực quan hóa trên nền tảng điện toán đám mây. Cụ thể:

+ Xây dựng pipeline dữ liệu tự động

- Tự động hóa quá trình ETL (Extract - Transform - Load) từ

MongoDB lên Cloud Storage và BigQuery.

- Đảm bảo dữ liệu được xử lý, lưu trữ đúng định dạng, có thể truy vấn và mở rộng.

+ Tích hợp các công cụ xử lý dữ liệu hiện đại

- Sử dụng Python để viết script xử lý, trích xuất dữ liệu và upload lên GCP.

- Tích hợp dbt để xây dựng mô hình dữ liệu, kiểm thử và quản lý schema.

+Trực quan hóa dữ liệu bằng dashboard chuyên nghiệp

- Kết nối BigQuery với Looker để tạo các báo cáo, dashboard phân tích theo thời gian, theo khu vực địa lý, doanh thu, hiệu suất sản phẩm...

Hệ thống ổn định và có khả năng mở rộng

- Ứng dụng kiến trúc điện toán đám mây đảm bảo tính linh hoạt, khả năng mở rộng và độ tin cậy của hệ thống dữ liệu.

- Triển khai cơ chế giám sát, ghi log và cảnh báo tự động để đảm bảo pipeline hoạt động liên tục.

II. Các chức năng của hệ thống

1. Chức năng dành cho người dùng phân tích dữ liệu (Data Analyst/Viewer)

→ Truy cập dashboard và báo cáo trực quan

❖ Truy cập hệ thống dashboard (Looker) để xem các báo cáo và biểu đồ

phân tích dữ liệu.

❖ Tùy chọn bộ lọc theo thời gian, khu vực địa lý, sản phẩm, doanh thu để khai thác insight.

❖ Xem các bảng phân tích như:

➢ Doanh thu theo ngày/tháng/quý.

➢ Hiệu suất sản phẩm theo vùng miền.

➢ Hành vi người dùng và tương tác sản phẩm.

→ Tìm kiếm & trích xuất dữ liệu phục vụ báo cáo

- Truy vấn dữ liệu trên BigQuery bằng các câu lệnh SQL hoặc từ dashboard Looker.
- Trích xuất dữ liệu dưới định dạng CSV hoặc Google Sheets phục vụ báo cáo kinh doanh.

---

2. Chức năng dành cho kỹ sư dữ liệu (Data Engineer)

→ Thiết lập và quản lý pipeline ETL

- Thiết lập các script Python tự động trích xuất dữ liệu từ MongoDB.
- Chuyển đổi định dạng dữ liệu sang CSV, JSON hoặc Parquet.
- Tải dữ liệu lên Google Cloud Storage (GCS) và kích hoạt Cloud Function để đưa vào BigQuery.

→ Theo dõi và bảo trì pipeline

- Quản lý file log của các job ETL (thành công/thất bại).
- Cấu hình cảnh báo tự động qua email hoặc Slack khi có lỗi pipeline.

- Kiểm thử pipeline từ đầu đến cuối mỗi lần cập nhật hoặc mở rộng.

→ Quản lý mô hình dữ liệu (dbt)

- Viết và cập nhật các mô hình dữ liệu (fact, dimension) bằng SQL trong dbt.
- Kiểm thử dữ liệu và áp dụng best practices trong việc xây dựng schema.
- Tự động build lại bảng khi có dữ liệu mới hoặc thay đổi mô hình.

---

3. Chức năng dành cho quản trị viên hệ thống (Admin)

→ Quản lý tài nguyên trên GCP

- Tạo và cấu hình các dịch vụ: GCS, Compute Engine, BigQuery, Cloud Functions.
- Phân quyền truy cập cho các thành viên dự án theo nguyên tắc least privilege.
- Giám sát mức sử dụng tài nguyên và chi phí trên GCP.

→ Giám sát bảo mật và nhật ký hệ thống

- Thiết lập cơ chế xác thực và bảo mật truy cập vào GCS, BigQuery.
- Theo dõi nhật ký truy cập hệ thống và hành động người dùng để phát hiện bất thường.
- Kiểm tra và xử lý các sự kiện lỗi hoặc truy cập trái phép.

III. Công nghệ sử dụng

→ Google Cloud Platform (GCP) – Nền tảng hạ tầng chính

Dự án sử dụng GCP để triển khai toàn bộ hệ thống từ lưu trữ đến xử lý và trực quan hóa dữ liệu.

- Google Cloud Storage (GCS): Lưu trữ dữ liệu thô và dữ liệu đã xử lý dưới nhiều định dạng như CSV, JSON, Parquet.
- Compute Engine (VM): Triển khai máy ảo để chạy MongoDB và thực hiện các script ETL.
- BigQuery: Kho dữ liệu phân tích mạnh mẽ, dùng để lưu trữ dữ liệu ở tầng phân tích và phục vụ truy vấn phức tạp.
- Cloud Functions: Tự động hóa pipeline bằng cách kích hoạt các tác vụ ETL và tải dữ liệu mỗi khi có thay đổi trong GCS.

→ **MongoDB** – Lưu trữ dữ liệu gốc

MongoDB được dùng làm nơi chứa dữ liệu thô thu thập được. Cơ sở dữ liệu NoSQL này phù hợp với dữ liệu phi cấu trúc hoặc bán cấu trúc, dễ dàng mở rộng và linh hoạt khi thao tác dữ liệu.

→ **Python** – Xử lý dữ liệu & Xây dựng pipeline

Ngôn ngữ chính dùng để xây dựng các bước trong pipeline:

- Kết nối với MongoDB, trích xuất dữ liệu
- Tiền xử lý, làm sạch và chuyển đổi định dạng
- Tải dữ liệu lên GCS hoặc BigQuery
- Script tự động hóa, logging, kiểm thử pipeline

→ **ip2location-python** – Xử lý vị trí địa lý từ IP

Thư viện Python được sử dụng để ánh xạ địa chỉ IP thành thông tin địa lý (quốc gia, thành phố,…), phục vụ phân tích theo vùng miền.

→ **dbt (Data Build Tool)** – Mô hình hóa và kiểm thử dữ liệu

Dùng để xây dựng mô hình dữ liệu trong BigQuery:

- Tạo bảng `dimension` và `fact`
- Viết các truy vấn SQL chuẩn hóa dữ liệu
- Kiểm thử, kiểm tra tính nhất quán
- Ghi chú tài liệu schema rõ ràng

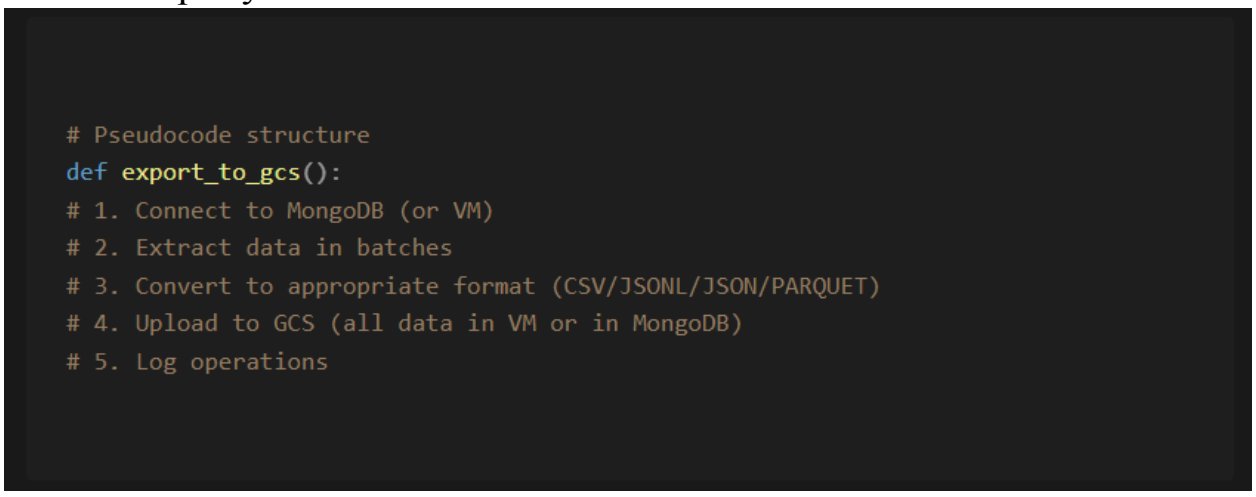
→ **Looker** – Trực quan hóa và phân tích dữ liệu

Công cụ BI để xây dựng dashboard phân tích:

- Biểu đồ doanh thu, xu hướng thời gian
- Phân tích theo vị trí địa lý, sản phẩm, hành vi người dùng
- Hỗ trợ đưa ra insight có giá trị từ dữ liệu

IV. Kế hoạch thực hiện dự án

-Tuần 1 - 2 của tháng 4: Xây dựng quy trình ETL & xuất dữ liệu

- → **Thực hiện quy trình xuất dữ liệu (ETL)**
  - Viết script Python để:
- 

```
# Pseudocode structure
def export_to_gcs():
# 1. Connect to MongoDB (or VM)
# 2. Extract data in batches
# 3. Convert to appropriate format (CSV/JSONL/JSON/PARQUET)
# 4. Upload to GCS (all data in VM or in MongoDB)
# 5. Log operations
```

  - Kết nối đến MongoDB (hoặc dữ liệu trong VM)
  - Trích xuất dữ liệu theo batch

- Chuyển đổi dữ liệu sang định dạng phù hợp (CSV / JSON / JSONL / PARQUET)
- Tải dữ liệu lên Google Cloud Storage (GCS)
- Ghi log quá trình thao tác

- **→ Xử lý lỗi và kiểm thử**
  - Bổ sung xử lý lỗi (try-except)
  - Ghi log chi tiết quá trình ETL
  - Kiểm thử với dữ liệu mẫu
- ---

-Tuần 3 - 4 của tháng 4: Tích hợp BigQuery & Kiểm thử tổng thể

- **→ Tích hợp với BigQuery**
  - Tạo dataset trong BigQuery
- ```
  python
  Copy
  # Pseudocode structure
  def trigger_bigquery_load(event, context):
  # 1. Detect new file in GCS
  # 2. Start BigQuery load job
  # 3. Log results
  ```
  - Xác định schema cho bảng dữ liệu
  - Viết script tải dữ liệu từ GCS vào BigQuery (raw layer)
  - Cài đặt Cloud Function tự động kích hoạt khi có file mới:
  - Cloud Function phát hiện file mới trên GCS
  - Kích hoạt job tải dữ liệu lên BigQuery
  - Ghi log kết quả tải dữ liệu
- **→ Kiểm thử và giám sát**
  - Kiểm thử toàn bộ pipeline đầu-cuối
  - Thiết lập cảnh báo (alerts) khi lỗi hoặc chậm
  - Viết tài liệu mô tả pipeline & hướng dẫn sử dụng
- ---

- Tuần 5 – 6 đầu tháng 5: Hoàn thiện, Kiểm thử & Triển khai

→ Hoàn thiện các tính năng nâng cao
  ```

● Tối ưu hóa script ETL: cải thiện hiệu suất tải dữ liệu lớn.
● Bổ sung cơ chế xử lý dữ liệu không hợp lệ hoặc sai định dạng.
● Triển khai hệ thống phân quyền truy cập vào dữ liệu BigQuery.

→ Kiểm thử toàn diện
● Kiểm thử toàn bộ pipeline từ MongoDB → GCS → BigQuery (manual & automated test).
● Kiểm thử Cloud Function trigger, xử lý lỗi và retry.
● Đánh giá hiệu suất tải và xử lý dữ liệu (performance test).

→ Tổng kết, viết tài liệu hướng dẫn & bảo trì
● Viết tài liệu hướng dẫn sử dụng pipeline, cài đặt và mở rộng.
● Xây dựng checklist bảo trì hệ thống định kỳ.
● Lên kế hoạch mở rộng pipeline (hỗ trợ thêm nguồn dữ liệu, chuẩn hóa dữ liệu,…).

●