WHAT / WHY / HOW

DECLARATION

ACKNOWLEDGEMENT

TABLE OF CONTENT

LIST OF ABBREVIATIONS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

I/ Introduction

1.      Overview (Include related work = literature review)

* Data lake
  + Là gì
  + Tại sao lại cần
  + Nếu ko có thì nnao
  + Các data lake phổ biến
  + Các nguồn dữ liệu lớn phổ biến (sentinel, landsat, dữ liệu vệ tinh của Nhật …)
* ETL
  + Là gì
  + Tại sao cần
  + Nếu ko có thì nnao
  + Một số ví dụ về ETL phổ biển
* Purpose: ULake hiện tại thiếu ETL cho landsat =>> Objective

2.      Objective: xây dựng 1 microservice cho data ingestion ingest dùng ETL process =>> Sẽ giúp ích cho các ứng dụng phân tích và giúp quá trình thu thập dữ liệu cho các machine learning model thuận tiện hơn

II/ Material and Method

1. Tools:

* Quarkus framework
  + Là gì
  + Why: phục vụ tốt khi làm việc với dữ liệu và tương thích với data lake hiện tại
  + Cụ thể ở project: include extension, code style and structure
* Postgresql
  + Là gì
  + Why: giúp việc lưu trữ dễ dàng
  + How: Nếu ko có thì so sánh với usual mysql
* gdal\_translate
  + Là gì
  + Tại sao cần: cho bước Transform
  + Nếu ko có thì nnao

1. Dataset:
   1. Mô tả dữ liệu vệ tinh
      1. Landsat/Sentinel/Ảnh vệ tinh nói chung là gì?
      2. Tại sao cần?
      3. Nếu ko có thì nnao?
   2. Mô tả nguồn dữ liệu mình dùng
      1. Copernicus là gì? Bao gồm cả Landsat/Sentinel
      2. Tại sao cần: mục đích cụ thể của các vệ tinh từ Copernicus
         1. Trong project dùng landsat hay sentinel, vệ tinh gì, theo dõi dữ liệu của vùng nào, với mục đích gì
      3. Nếu ko có thì sao?
2. Method
   1. Microservice
      1. Là gì (cụ thể ở đây là package usth.m1)
      2. Tai sao lai can microservice
      3. Nếu ko có thì phải dùng cgi thay thế? Bất tiện hơn nnao
   2. Data lake
      1. Loại data lake mình dùng là gì: Ulake
      2. Mô tả figure của data lake (mail thầy gửi)
      3. Chỉ ra microservice của mình nằm ở chỗ nào
   3. ETL microservice flow:
      1. Mô tả luồng chung: pull => save locally => upload 🡺 User requirements và use case diagrams (UML)???
      2. Extract (pull process)
         1. Setup token cho mỗi lần request
         2. Dữ liệu extract: ảnh true-color, ảnh full band, metadata
         3. Khi lưu local: ảnh theo folder, metadata vào postgresql + time info + directory info
         4. Sequence diagram?
      3. Transform (preprocess: downsize image)
         1. Problem: raw images can be very large => need a downsize technique to be able to upload
         2. Nếu ko có thì quy trình sẽ phức tạp hơn: lưu ảnh ở cloud và chỉ save link ở lake
         3. Sequence diagram?
      4. Load (upload to lake)
         1. Retrieve token cho mỗi lần call
         2. Dùng file/folder microservice (nhắc về bất lợi Ulake nên ko dùng đc Object)
         3. Thông tin lưu: ảnh kèm 1 số thông tin cơ bản (nhắc về bất lợi Ulake nên ko upload đc metadata)
         4. Sequence diagram?

III/ Results & Discussion

* Kết quả:
  + Hoạt động nnao?
  + Extract đc loại dữ liệu gì
  + Dữ liệu preprocess thành công ko? Kết quả đầu ra preprocess nnao?
  + Upload dữ liệu vào datalake đc ko? Du lieu gi?

IV/ Conclusion & Future work

1. Overview: Paraphrase lại intro
2. Future works:
   1. Tạo log để theo dõi quá trình ETL
   2. Tạo cronjob chạy hàng ngày cho ETL để theo dõi 1 vài khu vực nhất định phục vụ nghiên cứu
   3. Thêm các bước preprocess để đảm bảo dữ liệu sạch: deduplication, avoid các ảnh trống (thường các ảnh này sẽ chỉ toàn màu đen. Dẫn chứng 1 ảnh).

V/ References

* Tầm 15 mục, chủ yếu sẽ nằm ở intro