TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

**VIỆN CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

----- 🙡 🕮 🙣 -----



**BÁO CÁO BÀI TẬP LỚN**

**LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU LỚN**

***Đề tài:***

Nhóm sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Tô Hoài Sơn | 20183823 |
| 2 | Ngô Văn Quang | 20183813 |
| 3 | Đỗ Hoàng Dũng | 20183717 |
| 4 | Lương Trọng Trí | 20183641 |

**Hà Nội, ngày 20 tháng 12 năm 2021**

# MỤC LỤC

[MỤC LỤC 2](#_Toc92339573)

[MỞ ĐẦU 3](#_Toc92339574)

[CHƯƠNG 1. TỔNG QUAN HỆ THỐNG 4](#_Toc92339575)

[CHƯƠNG 2. THU THẬP DỮ LIỆU 5](#_Toc92339576)

[2.1. Nguồn gốc dữ liệu 5](#_Toc92339577)

[2.2. Phương pháp lấy dữ liệu 7](#_Toc92339578)

[2.3. Kết quả thu được 9](#_Toc92339579)

[CHƯƠNG 3. LƯU TRỮ DỮ LIỆU 11](#_Toc92339580)

[CHƯƠNG 4. XỬ LÝ DỮ LIỆU 12](#_Toc92339581)

[CHƯƠNG 5. TRỰC QUAN HÓA DỮ LIỆU 13](#_Toc92339582)

[CHƯƠNG 6. TỔNG KẾT, KẾT LUẬN VÀ ĐÁNH GIÁ 14](#_Toc92339583)

[6.1. Bảng phân chia công việc của các thành viên và đánh giá 14](#_Toc92339584)

[6.2. Những kết quả thu được 15](#_Toc92339585)

[6.3. Khó khăn 15](#_Toc92339586)

[TÀI LIỆU THAM KHẢO 16](#_Toc92339587)

# MỞ ĐẦU

Dữ liệu lớn (Big Data) là các tập dữ liệu có khối lượng lớn và phức tạp, độ lớn đến mức các phần mềm xử lý dữ liệu truyền thống không có khả năng thu thập, quản lý và xử lý dữ liệu trong một khoảng thời gian hợp lý. Tuy nhiên, Big Data lại chứa trong mình rất nhiều thông tin quý giá mà nếu trích xuất thành công, nó sẽ đem lại nguồn tri thức khổng lồ, giúp ứng dụng cho nhiều bài toán của khoa học, kĩ thuật vào đời sống. Vì khối dữ liệu quá lớn nên việc triển khai Big Data sẽ gặp những trở ngại bao gồm thu nhận dữ liệu, lưu trữ dữ liệu, tìm kiếm.

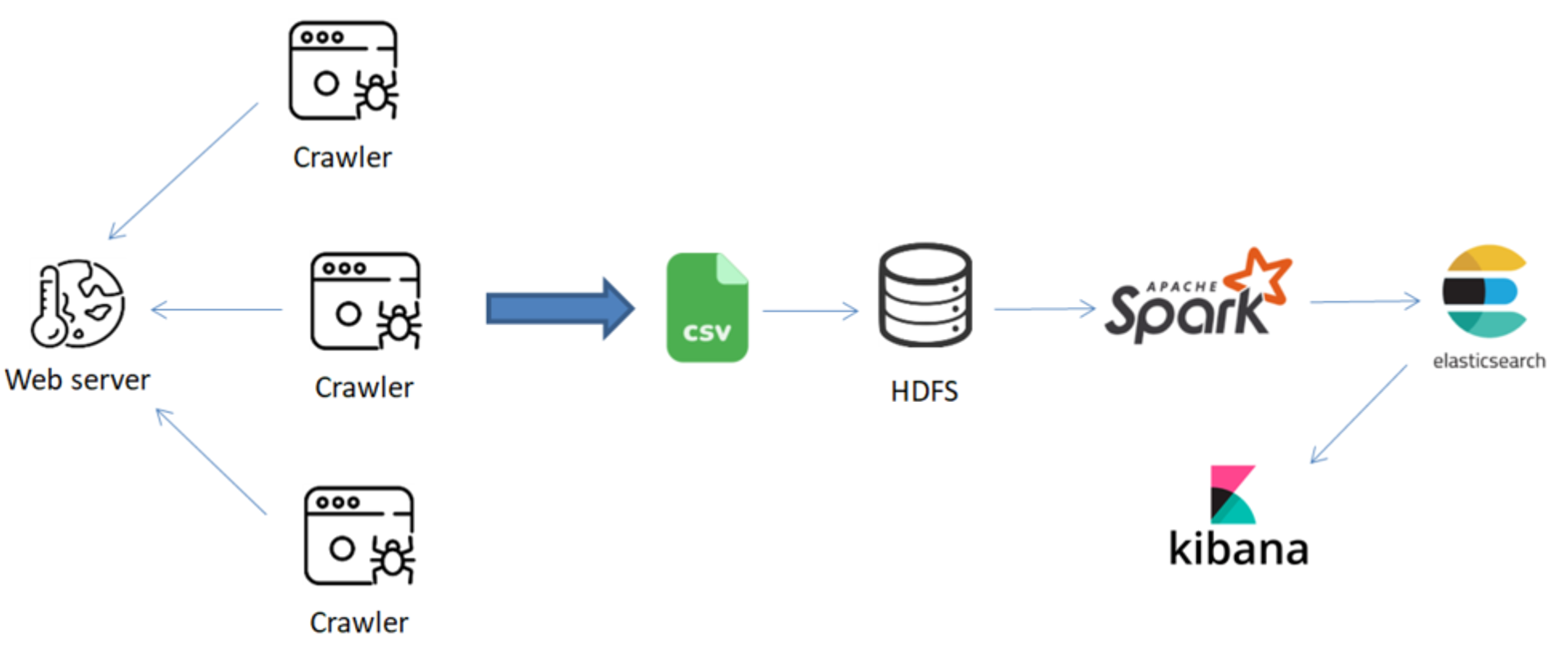
Để xử lý được nguồn dữ liệu khổng lồ này, ta có thể nghĩ ngay tới việc phải có một máy tính có cấu hình mạnh để xử lý, tuy nhiên, điều này là bất khả thi do gặp giới hạn về công nghệ phần cứng cũng như khả năng scale theo chiều dọc không phải lúc nào cũng đảm bảo về chi phí. Do đó các giải pháp scale theo chiều ngang, cụ thể đó là việc kết nối nhiều máy tính để tận dụng sức mạnh tính toán song song.

Với các vấn đề nêu trên thì hiện nay đã có rất nhiều công nghệ ra đời để giải quyết việc lưu trữ và xử lý song song này, ví dụ như Apache Hadoop, Apache Spark, … Nhận thức được những điều này, nhóm chúng em quyết định triển khai một hệ thống nhỏ có chức năng thu thập, xử lý, lưu trữ và trực quan hóa dữ liệu.

Cụ thể, nhóm chúng em hướng tới việc sử dụng các công cụ như Hadoop, Spark, … để lưu trữ và xử lý dữ liệu về khí hậu crawl được từ nguồn [*https://en.tutiempo.net/climate*](https://en.tutiempo.net/climate) - đây là nguồn dữ liệu về khí hậu trên toàn thế giới.

# CHƯƠNG 1. TỔNG QUAN HỆ THỐNG

Sơ đồ tổng quan hệ thống:

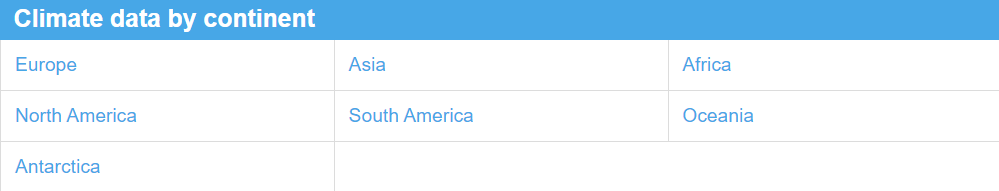
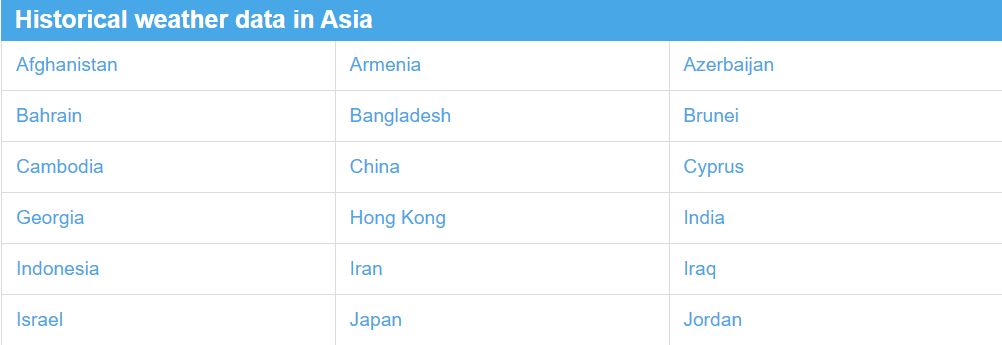
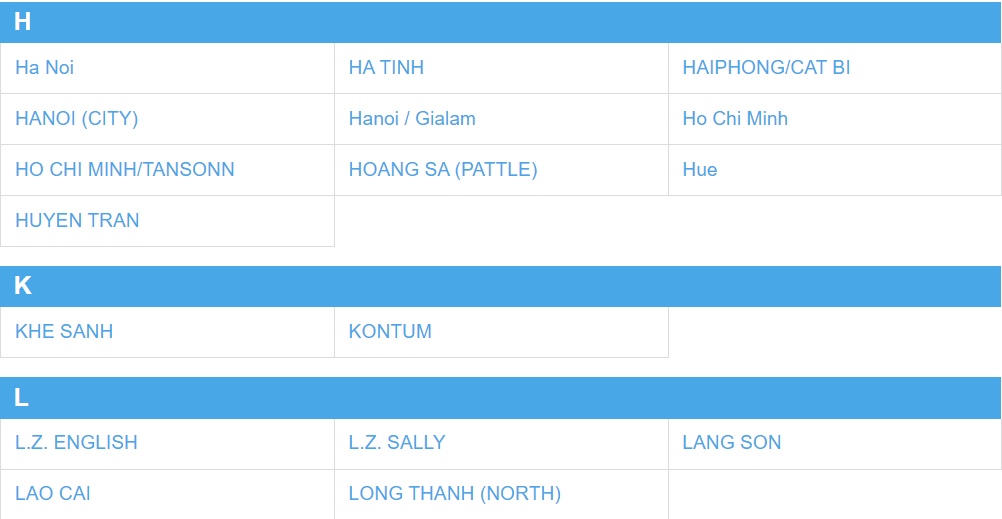
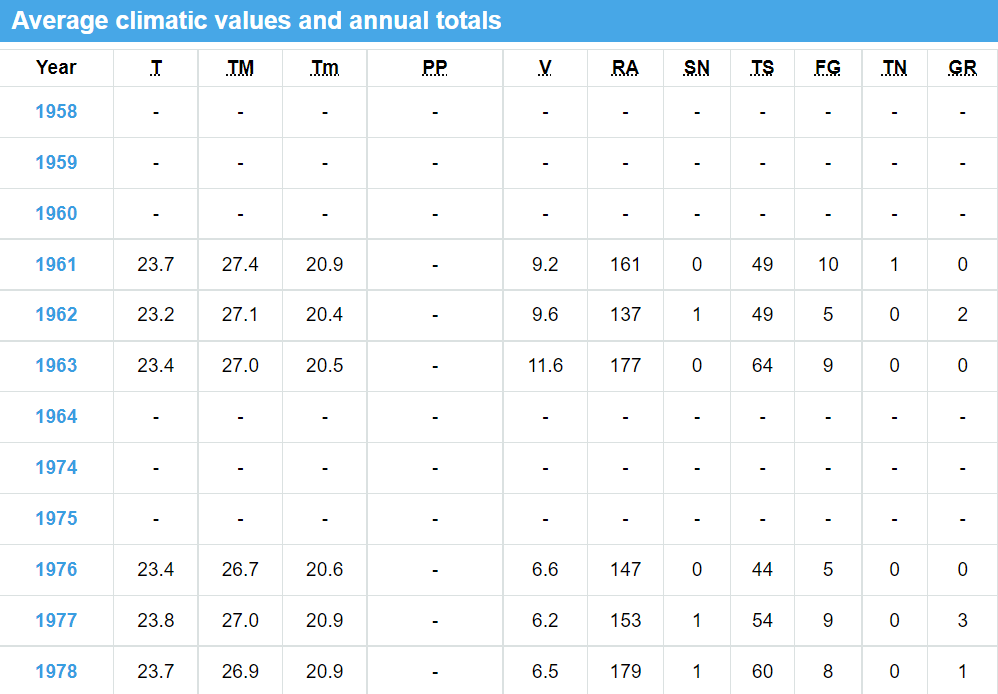
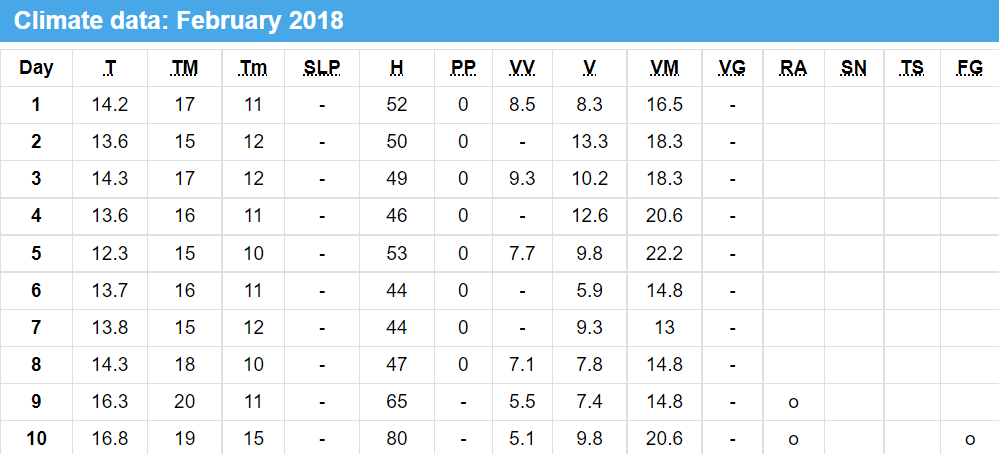
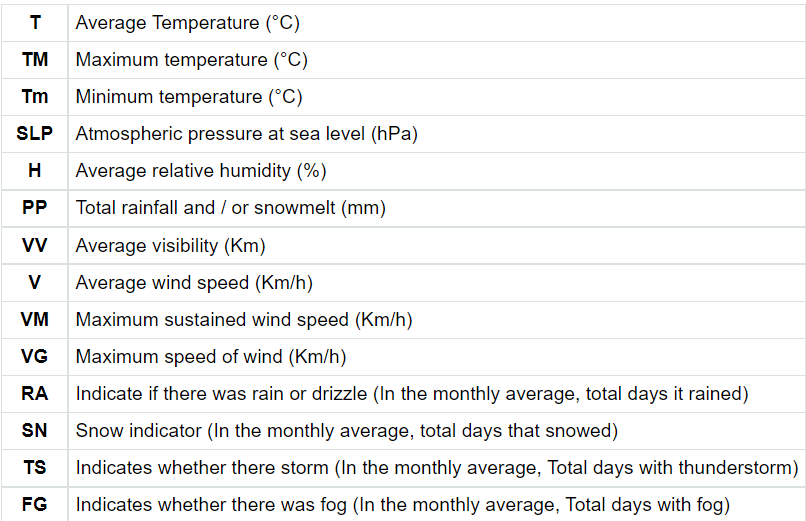


* Giải thích các thành phần trong hệ thống:
  + Web server: nơi ta sẽ crawl dữ liệu thô – dữ liệu dạng html
  + Crawler: bot crawl dữ liệu, thực hiện một vài thao tác xử lý dữ liệu để đưa về các file .csv
  + HDFS: Hadoop Distributed File System – hệ thống lưu trữ file phân tán
  + Spark: engine cung cấp các phương thức giúp xử lý dữ liệu lớn
  + Elasticsearch: search engine có tích hợp một NoSql database, hỗ trợ lưu trữ và tìm kiếm dữ liệu nhanh chóng
  + Kibana: công cụ trực quan hóa dữ liệu trong Elasticsearch, hoạt động tốt với Elasticsearch (cùng tech stack)
* Luồng chính của hệ thống:

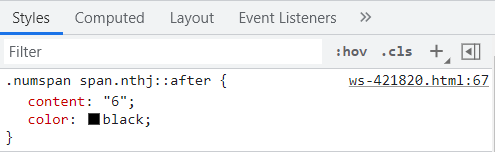
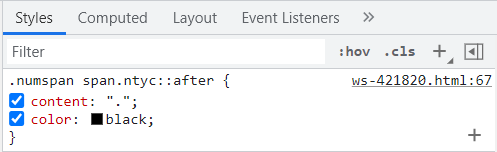
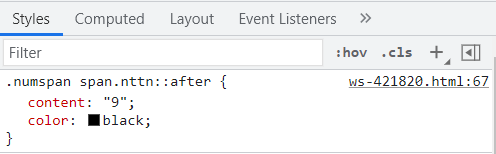
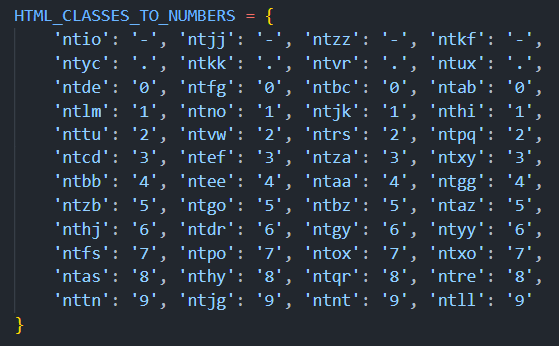
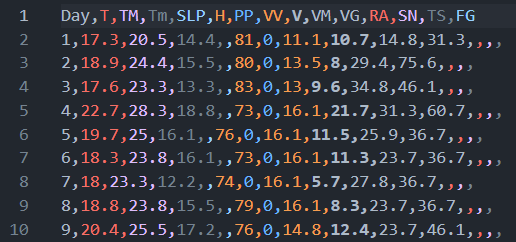
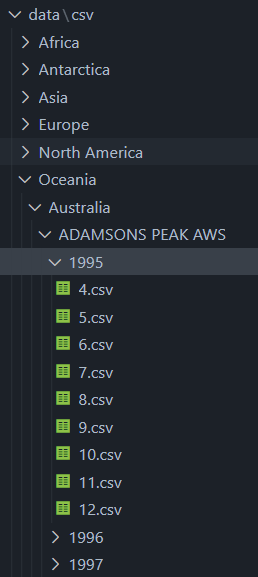
Các crawlers lấy dữ liệu thô từ Web server, được các file thô html 🡪 Các crawlers này thực hiện một vài thao tác tiền xử lý, sau đó gộp ra file .csv cuối cùng 🡪 Lưu .csv trong HDFS 🡪 Spark lấy dữ liệu .csv này ra và xử lý 🡪 Spark lưu dữ liệu đã xử lý xong vào Elasticsearch. Sau cùng, Kibana có thể lấy dữ liệu từ Elasticsearch và trực quan hóa lên View.

# CHƯƠNG 2. THU THẬP DỮ LIỆU

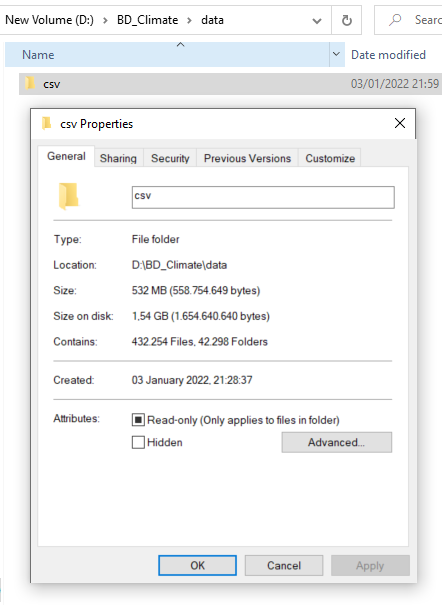
## 2.1. Nguồn gốc dữ liệu

* Dữ liệu được lấy từ web platform lưu trữ lịch sử khí hậu trên toàn cầu từ khoảng giữa thế kỉ XX tới bây giờ: [*https://en.tutiempo.net/climate*](https://en.tutiempo.net/climate)
* Dữ liệu được lưu trữ theo từng danh mục, cụ thể, thì sẽ được lưu trữ theo lục địa / quốc gia (đất nước, vùng lãnh thổ) / thành phố (bang, tỉnh) / năm / tháng.
  + Dữ liệu theo *lục địa (vùng lãnh thổ)*. Ví dụ:  
    
  + Dữ liệu theo *quốc gia*. Ví dụ:  
    
  + Dữ liệu theo *thành phố* *(bang, tỉnh)*. Ví dụ:  
    
  + Dữ liệu theo *năm*. Ví dụ:  
    
  + Dữ liệu theo *tháng*. Ví dụ:  
    
* Một bảng dữ liệu sẽ có dạng sau:  
  
* Bảng giải thích các kí hiệu trong bảng dữ liệu:  
  

## 2.2. Phương pháp lấy dữ liệu

* Dữ liệu trong trang được lưu trong dạng bảng, thực chất là các *rows* trong phần tử *table* của trang, thêm nữa, trang này cũng không cung cấp một *public api* cho việc *get* dữ liệu.
* Với những điều kiện trên, nhóm sẽ tạo các *crawlers* để kéo dữ liệu thô (dữ liệu dạng *html*) đồng thời xử lý và lưu trữ tại local.
* Các *crawlers* có nhiệm vụ tạo requests tới trang web, nhận dữ liệu thô trả về, xử lý và bóc tách các *rows* dữ liệu và xuất dữ liệu ra các files .csv tương ứng.
* Ban đầu nhóm sử dụng 1 con crawler, tuy nhiên do lượng dữ liệu khá nhiều, thời gian gửi – nhận requests khá lâu, nên nhóm sẽ chia lượng dữ liệu cần lấy thành nhiều phần và thực hiện chạy nhiều crawlers cùng một lúc. Ví dụ:  
  crawl\_asia.py 🡪 crawl dữ liệu của Asia  
  crawl\_europe.py 🡪 crawl dữ liệu của Europe
* Dữ liệu sau đó được làm sạch (loại bỏ kí tự lạ, kí tự đặc biệt như ‘-‘, ‘o’ hay ‘&nbsp’). Đặc biệt, trong quá trình phân tích cấu trúc trang web, nhóm nhận thấy trang có encode một số ô dữ liệu bằng một html pseudo element và bọc trong thẻ <span>. Ví dụ <td> dưới dây thể hiện số 6.9:  
    
    
    
    
    
    
    
    
  🡪 Nhóm sẽ tiến hành map các html class sang số được encode tương ứng và xây dựng một bảng decode và lưu trong một biến, sau đó thực hiện thay thế các lớp này bằng contents tương ứng. Ảnh biến chứa bảng decode:  
  
* Sau khi tiền xử lý và chuẩn hóa về file csv, ta có ví dụ về một file chứa dữ liệu về một tháng trong một năm của một thành phố:  
  
* Mỗi file .csv trên sẽ được lưu trong một thư mục phân cấp tương ứng theo dạng: *data/csv/{lục địa}/{quốc gia}/{thành phố}/{year}/{month}.csv.* Ví dụ:  
  

## 2.3. Kết quả thu được

* Sau khi crawl và lưu trữ vào *data/csv*, nhóm thu được lượng dữ liệu vào khoảng 532MB 🡪 đạt yêu cầu đề ra ban đầu là *tối thiểu 512 MB*  
  

# CHƯƠNG 3. LƯU TRỮ DỮ LIỆU

# CHƯƠNG 4. XỬ LÝ DỮ LIỆU

# CHƯƠNG 5. TRỰC QUAN HÓA DỮ LIỆU

# CHƯƠNG 6. TỔNG KẾT, KẾT LUẬN VÀ ĐÁNH GIÁ

## 6.1. Bảng phân chia công việc của các thành viên và đánh giá

|  |  |  |
| --- | --- | --- |
| **Công việc** | **Mô tả chi tiết** | **Thành viên thực hiện** |
| Tìm ý tưởng thực hiện |  | Cả team |
| Lên kế hoạch thực hiện và phân chia công việc |  | Lương Trọng Trí |
| Thiết kế hệ thống | Thiết kế tổng quan | Cả team |
| Lựa chọn thành phần triển khai |
| Thu thập dữ liệu | Tìm nguồn dữ liệu | Đỗ Hoàng Dũng |
| Crawl dữ liệu thô (code crawlers) |
| Tiền xử lý dữ liệu |
| Gộp các file dữ liệu | Lương Trọng Trí |
| Cài đặt môi trường | Cài đặt cụm Hadoop | Tô Hoài Sơn |
| Cài đặt cụm Spark |
| Cài đặt ElasticSearch | Ngô Văn Quang |
| Cài đặt Kibana |
| Khác (tính toán dung lượng, cấu hình, tài nguyên máy của các thành viên, góp ý, fix bugs, …) | Cả team |
| Thao tác với dữ liệu trong hệ thống | Lưu trữ dữ liệu với Hadoop | Cả team |
| Xử lý dữ liệu với Spark | Lương Trọng Trí |
| Trực quan hóa dữ liệu với Kibana | Ngô Văn Quang |
| Viết báo cáo tổng hợp |  | Đỗ Hoàng Dũng |
| Thuyết trình |  | Lương Trọng Trí |
| Tổng kết và đánh giá kết quả |  | Cả team |

**Đánh giá:**

* Các thành viên đều hoàn thành tốt công việc được giao
* Thành viên trong nhóm luôn tham gia hỗ trợ lẫn nhau kể cả phần việc đó không phải được phân công cho mình
* Mọi người đều tham gia đầy đủ các cuộc họp nhóm và tổng kết đánh giá kết quả

## 6.2. Những kết quả thu được

* Hiểu được cơ bản luồng bài toán Big Data
* Cải thiện kĩ năng giao tiếp và làm việc nhóm (cùng cài đặt môi trường, fix bugs, cấu hình, ...)
* Nắm được một số kĩ thuật trong crawl
  + Bóc tách dữ liệu từ dữ liệu thô html
  + Xử lý các vấn đề liên quan tới connection, timeout
  + Sinh nhiều crawlers để đẩy nhanh việc crawl dữ liệu
* Thêm các kĩ năng về cấu hình cho server (Hadoop, Spark, …), làm việc với các file configs, các thao tác dòng lệnh
* Vận dụng được kiến thức đã học về lưu trữ trong HDFS, xử lý dữ liệu với Spark và lưu trữ với Elasticsearch
* Trực quan hóa và khám phá dữ liệu với Kibana

## 6.3. Khó khăn

* Cấu hình máy của các thành viên trong nhóm tương đối thấp (các thành viên bị giới hạn bởi lượng RAM 8GB, dung lượng ổ cứng thấp) 🡪 trở ngại lớn nhất
* Đường truyền mạng thường xuyên chập chờn, kém dẫn tới việc không triển khai được VPN
* Dịch bệnh Covid-19 🡪 không gặp mặt trực tiếp để trao đổi cũng như ngồi lại để cùng làm bài
* Việc cài đặt môi trường đôi khi còn nhiều khó khăn vì thiếu kinh nghiệm và môi trường giữa các máy là khác nhau

# TÀI LIỆU THAM KHẢO

1. TS. Đào Thành Chung. Bộ slides bài giảng môn *Lưu trữ và xử lý dữ liệu lớn*
2. Spark documentation. [*https://spark.apache.org/docs/latest/*](https://spark.apache.org/docs/latest/)
3. Hadoop documentation. [*https://hadoop.apache.org/docs/r1.2.1/hdfs\_user\_guide.html*](https://hadoop.apache.org/docs/r1.2.1/hdfs_user_guide.html)
4. Elastic documentation. [*https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html*](https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html)
5. Elastic documentation. [*https://www.elastic.co/guide/en/kibana/current/index.html*](https://www.elastic.co/guide/en/kibana/current/index.html)
6. Pandas documentation. [*https://pandas.pydata.org/docs/*](https://pandas.pydata.org/docs/)
7. BeautifulSoup documentation. [*https://beautiful-soup-4.readthedocs.io/en/latest/*](https://beautiful-soup-4.readthedocs.io/en/latest/)