

**Tên đề tài:** Xây dựng ứng thu thập, trực quan hóa và dự báo dữ liệu giá chứng khoán trên nền tảng blockchain.

(\*) *Lưu ý:* Khi đọc tài liệu nghiên cứu, sinh viên cần ghi lại các công nghệ, thành phần, mô hình, kiến trúc được sử dụng trong đồ án

## Mô tả

**Giai đoạn 1:** Xây dựng dịch vụ Web Crawling để thu thập dữ liệu các mã chứng khoán. Dịch vụ có thể chạy trên máy localhost hoặc deploy lên máy ảo AWS (AWS Free tier)

- *Mục tiêu:* Tất cả thông tin dữ liệu liên quan về các mã chứng khoán có thể có được (vài GB) để trở thành Big Data

### Tài liệu tham khảo

1. Khái niệm Web Crawler: <https://www.scrapingbee.com/blog/crawling-python/>
2. Ví dụ: <https://www.digitalocean.com/community/tutorials/how-to-crawl-a-web-page-with-scrapy-and-python-3>
3. Xây dựng Scrapy với Flask Rest API (đọc các bài liên quan): <https://dingyuliang.me/scrapy-build-scrapy-flask-rest-api-1/>
4. Tìm hiểu Swagger: <https://viblo.asia/p/co-ban-ve-swagger-oOVYjOV58W>
5. Xây dựng API bằng Flask và Swagger: <https://viblo.asia/p/dung-api-bang-flask-voi-giao-dien-nguoi-dung-swagger-E375zLPbZGW> hoặc <https://www.imaginarycloud.com/blog/flask-python/>

**Giai đoạn 2:** Xây dựng Data Lake trên nền AWS (Data Lake on AWS),

- *Mục tiêu:* Triển khai thành công hệ thống lưu trữ trên AWS và hiểu được sự kết hợp của các thành phần trong đám mây AWS

*Lưu ý:* Tạo tài khoản AWS Free Tier (nếu chưa có)

- Phần này sẽ tốn kém chi phí cho cloud, sinh viên cần chịu khó đầu tư để nâng cao kiến thức và đọc kỹ các chính sách của AWS để không phải bị tính phí oan uổng.
- Sinh viên đưa ra giải pháp sử dụng Data Warehouse hay Data Lake dựa trên các tài liệu tham khảo

### Tài liệu tham khảo

1. Tìm hiểu về Data Warehouse và Data Lake trong thực tế: <https://panoply.io/data-warehouse-guide/data-warehouse-vs-data-lake/>
2. So sánh quá trình xử lý của Data Lake và Data Warehouse: <https://samuelsum.com/data-lake-versus-data-warehouse/>
3. Xây dựng Metadata Catalog trên AWS: <https://www.slideshare.net/AmazonWebServices/building-a-metadata-catalog-for-your-data-lakes-using-amazon-elasticsearch-service-july-2017>
4. Xây dựng dịch vụ Data Lake trên AWS: [slide](#) hoặc [https://uto.asu.edu/sites/default/files/modernserverlessdatalake\\_asu\\_innov8.pdf](https://uto.asu.edu/sites/default/files/modernserverlessdatalake_asu_innov8.pdf)

5. Cài đặt kiến trúc Data Lake trên AWS: <https://www.srijan.net/resources/blog/setting-up-a-data-lake-architecture-with-aws>
6. Giải pháp Data Lake trên AWS: <https://aws.amazon.com/solutions/implementations/data-lake-solution/> và <https://docs.aws.amazon.com/whitepapers/latest/building-data-lakes/data-cataloging.html>
7. Các bài đọc thêm
  - a. Elastic Search: <https://dingyuliang.me/elasticsearch-import-data-csv-using-python/>
  - b. Modern EDW: [slide](#) hoặc [https://d1.awsstatic.com/products/Redshift/Amazon%20Data%20Warehouse%20EBook\\_111518.pdf](https://d1.awsstatic.com/products/Redshift/Amazon%20Data%20Warehouse%20EBook_111518.pdf)
  - c. EDW trên AWS: [slide](#) hoặc <https://d1.awsstatic.com/whitepapers/enterprise-data-warehousing-on-aws.pdf>
  - d. Big Data: [Slide](#)
  - e. Sinh viên tự tìm hiểu thêm về Big Data, ETL hoặc ELT ,OLTP và OLAP

### **Giai đoạn 3:** Trực quan hóa dữ liệu chứng khoán

*Lưu ý:* Sinh viên có thể sử dụng Python hoặc các công cụ khác để trực quan như Power BI, Tableau... Khuyến cáo dùng Python cho linh hoạt, nếu đã biết rõ về MS Power BI thì có thể sử dụng Power BI xây dựng Dashboard để nhúng vào web.

- *Mục tiêu:* Sinh viên thiết kế các dữ liệu cần trực quan và biểu đồ phù hợp với nghiệp vụ chứng khoán
  - o *Gợi ý:*
    - *Phần sử dụng miễn phí:* Sinh viên cần khảo sát, tìm hiểu xem các nhà đầu tư quan tâm giá trị dữ liệu nào, thường nhìn biểu đồ nào để quyết định xuống tiền đầu tư (đặt lệnh)
    - *Phần sử dụng thương mại (tính phí):* Xây dựng biểu đồ so sánh các mã chứng khoán khác nhau dựa theo
      - Số lượng mã dùng so sánh
      - Khoảng thời gian dài nhất để so sánh
      - Dự báo với mô hình LSTM
      - Dự báo với mô hình ARIMA
      - Khai thác đặc tính mùa (chu kỳ) của mã chứng khoán: mô hình SARIMA
      - ..... có thời gian nhiều thì hỏi thầy nhiều thuật toán khác để cài đặt

Tham khảo

1. Ứng dụng mvc với Flask Python: <https://python.plainenglish.io/flask-crud-application-using-mvc-architecture-3b073271274f> hoặc <https://towardsdatascience.com/creating-restful-apis-using-flask-and-python-655bad51b24> hoặc <https://python-rest-framework.readthedocs.io/en/latest/introduction.html#architecture>

2. Có thể tìm hiểu thêm Django MVC: <https://viblo.asia/p/tim-hieu-ve-django-framework-ho-tro-python-trong-lap-trinh-web-QpmlexbkZrd>

**Giai đoạn 4:** Xây dựng quyền truy cập phần thương mại dựa trên tài nguyên NFT kết hợp ví Meta Mask.

*Lưu ý:* Nếu làm kịp giai đoạn 4 thì sẽ giữ nguyên tên đề tài, nếu không kịp sẽ đổi tên đề tài cho phù hợp.

- *Mục tiêu:* Hoàn thành việc sử dụng mua bán tài sản NFT để có quyền sử dụng tính năng thương mại của ứng dụng.

*Tài liệu tham khảo*

1. Sinh viên tự tìm hiểu Blockchain, NFT và Meta Mask Wallet
2. ETH cho lập trình viên Python: <https://ethereum.org/en/developers/docs/programming-languages/python/>
3. Cơ bản tạo NFT với tool: <https://www.theverge.com/22809090/nft-create-opensea-rarible-cryptocurrency-ethereum-collectibles-how-to>
4. Tạo NFT với Python: <https://betterprogramming.pub/create-your-own-nft-collection-with-python-82af40abf99f>
5. Xây dựng mẫu ứng dụng tài chính phân tán: <https://itnext.io/develop-your-first-decentralized-e-commerce-application-with-python-flask-and-metamask-96d9344e415d>
6. Đưa tài sản NFT lên OpenSea: <https://www.freecodecamp.org/news/how-to-make-an-nft-and-render-on-opensea-marketplace/>