

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM
KHOA: HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM



BÁO CÁO ĐỒ ÁN
CHUYÊN NGÀNH CÔNG NGHỆ PHẦN MỀM
CÔNG NGHỆ DỮ LIỆU LỚN

NGHIÊN CỨU THU THẬP PHÂN TÍCH DỮ LIỆU
VÀ ỨNG DỤNG THỰC TIỄN

Giảng viên hướng dẫn: **ThS. Cao Hữu Thanh Vũ**

Sinh viên thực hiện: **Lê Thị Bảo Yến**

Mã số sinh viên: **0850080057**

Lớp: **08_ĐH_CNPM**

Khoá: **08**

TP. Hồ Chí Minh, tháng 05 năm 2023.

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM
KHOA: HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM



BÁO CÁO ĐỒ ÁN
CHUYÊN NGÀNH CÔNG NGHỆ PHẦN MỀM
CÔNG NGHỆ DỮ LIỆU LỚN

NGHIÊN CỨU THU THẬP PHÂN TÍCH DỮ LIỆU
VÀ ỨNG DỤNG THỰC TIỄN

Giảng viên hướng dẫn: **ThS. Cao Hữu Thanh Vũ**

Sinh viên thực hiện: **Lê Thị Bảo Yến**

Mã số sinh viên: **0850080057**

Lớp: **08_ĐH_CNPM**

Khoá: **08**

TP. Hồ Chí Minh, tháng 05 năm 2023.

LỜI MỞ ĐẦU

Trong thời đại hiện nay, dữ liệu ngày càng phát triển với tốc độ chóng mặt. Các nguồn dữ liệu đa dạng từ mạng xã hội, trang web, thiết bị IoT và hệ thống thông tin đem lại một lượng dữ liệu khổng lồ và đa dạng. Việc thu thập, phân tích và quản lý dữ liệu trở thành một nhiệm vụ quan trọng và không thể thiếu trong môi trường kinh doanh và công nghiệp hiện đại.

Thu thập dữ liệu đóng vai trò quan trọng trong việc tạo ra thông tin quý giá và kiến thức. Dữ liệu được xem như một nguồn tài nguyên vô cùng quý giá và khả năng thu thập, xử lý, phân tích dữ liệu đúng lúc đã trở thành lợi thế cạnh tranh đáng kể.

Sự thu thập dữ liệu có thể giúp tăng cường sự hiểu biết về khách hàng, cải thiện trải nghiệm người dùng, tối ưu hóa quy trình sản xuất và phân phối, và tạo ra các sản phẩm và dịch vụ mới. Ngoài ra, dữ liệu còn đóng vai trò quan trọng trong việc nghiên cứu khoa học, phát triển công nghệ, dự báo thời tiết, phân tích y tế, và nhiều lĩnh vực khác.

Ứng dụng phân tích dữ liệu vào thực tiễn mang lại nhiều lợi ích đáng kể. Trong lĩnh vực kinh doanh, phân tích dữ liệu giúp tối ưu hóa chiến lược tiếp thị, nâng cao trải nghiệm khách hàng, dự báo xu hướng thị trường. Em đã quyết định thực hiện đề tài: **“Nghiên Cứu Thu Thập Phân Tích Dữ Liệu Và Ứng Dụng Thực Tiễn”** với sự hướng tới là tìm hiểu nghiên cứu một số phương pháp phân tích dữ liệu, ứng dụng vào thực tiễn.

LỜI CẢM ƠN

Lời đầu tiên cho em xin gửi lời cảm ơn ban giám hiệu Trường Đại học Tài nguyên và Môi trường TP. Hồ Chí Minh đã cung cấp môi trường học tập và nghiên cứu cho sinh viên. Em xin gửi lời cảm ơn đến toàn thể thầy cô khoa Hệ thống thông tin và Viễn thám, đã tạo điều kiện cho em có thể học hỏi tích lũy kiến thức và kỹ năng thực tế hơn và cách khắc phục trong việc phát triển các sản phẩm trong hiện tại và tương lai.

Em xin gửi lời cảm ơn chân thành đến giảng viên thầy ThS. Cao Hữu Thanh Vũ đã tận tình hướng dẫn giúp đỡ em trong suốt quá trình thực hiện, thầy đã chỉ dạy em những kiến thức mới và thực tế để em có đủ kiến thức và vận dụng chúng vào báo cáo đồ án này. Kính chúc thầy thật nhiều sức khỏe.

Trong quá trình thực hiện đề tài, em có những thiếu sót nhất định và kiến thức còn hạn chế. Em rất mong nhận được ý kiến đóng góp của thầy để em có thêm kinh nghiệm và tiếp tục hoàn thiện năng lực của mình.

Em xin chân thành cảm ơn!

LỜI CAM ĐOAN

Em xin cam đoan đề án này là sản phẩm nghiên cứu, tìm hiểu và phát triển của em, không sao chép.

Em xin chịu trách nhiệm về lời cam đoan của mình.

TP.HCM, ngày 28 tháng 5 năm 2023

Sinh viên thực hiện

NHẬN XÉT

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

....., ngày....tháng....năm.....
NGƯỜI NHẬN XÉT
(ký tên)

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	12
1.1. Giới thiệu về đề tài	12
1.2. Mục tiêu đề tài	12
1.3. Phạm vi đề tài	12
CHƯƠNG II: CƠ SỞ LÝ THUYẾT	13
2.1. Lý thuyết.....	13
2.1.1. Big Data	13
2.1.2. Xử lý phân tích dữ liệu	14
2.2. Kỹ thuật	15
2.2.1. Ngôn ngữ C#	15
2.2.2. Ngôn ngữ Python.....	16
2.2.3. Mô hình MVC	16
2.2.4. Spyder và Visual Studio Code.....	17
2.2.5. SQL Server	18
2.2.6. ASP.NET Core 3.1 và Entity Framework Core	18
2.2.7. Chart.js.....	19
2.2.8. Pyscript	20
2.2.9. Git và GitHub	20
CHƯƠNG III: CÀI ĐẶT THỬ NGHIỆM	21
3.1. Phương pháp nghiên cứu	21
3.2. Phương pháp thực hiện	22
3.2.1 Thu thập dữ liệu.....	22
3.2.2 Làm sạch dữ liệu.....	24
3.2.3 Trực quan dữ liệu	29
3.2.4 Ứng dụng	37
3.3 Một số nghiên cứu khác	44
3.3.1 Tạo tự động dữ liệu mẫu	44
3.3.2 Thêm dữ liệu tự động	48
3.4 Kết quả đạt được	50
CHƯƠNG IV: KẾT LUẬN.....	51
4.1. Kết luận.....	51
4.1.1. Đánh giá kết quả đạt được	51

4.1.2. Kiến thức đạt được	51
4.2. Hướng phát triển.....	51
TÀI LIỆU THAM KHẢO	53

DANH MỤC HÌNH ẢNH

Hình 3. 1 Trực quan với python	30
Hình 3. 2 Trực quan với ChartJs	33
Hình 3. 3 Trực quan với Pyscript	36
Hình 3. 4 Kết quả thu thập được của công cụ tự tạo	40

DANH MỤC BẢNG

Bảng 3. 1 Dữ liệu bản đầu	24
Bảng 3. 2 Dữ liệu đã sạch.....	28
Bảng 3. 3 Dữ liệu trùng	29

KÝ HIỆU CỤM TỪ VIẾT TẮT

ASP NET CORE 3.1	Active Server Pages .NET Core
SQL	Structure Query Language
CSS	Cascading Style Sheets
HTML	Hyper Text Markup Language
JS	JavaScript
JSON	JavaScript Object Notation
MVC	Model-View-Controller
OOP	Object Oriented Programming
SQL	Structure Query Language
VNĐ	Việt Nam đồng

CHƯƠNG 1: TỔNG QUAN

1.1. Giới thiệu về đề tài

Trong đề tài này, em sẽ tìm hiểu về Big Data và tầm quan trọng của phân tích dữ liệu. Chúng ta sẽ khám phá các phương pháp và công cụ phân tích dữ liệu hiện đại và tìm hiểu cách áp dụng chúng vào các lĩnh vực khác nhau trong thực tế. Qua đó, có thể hiểu rõ hơn về tiềm năng của Big Data và phân tích dữ liệu trong việc tạo ra giá trị và định hình sự phát triển của các tổ chức và xã hội trong thời đại số hóa ngày nay.

Đề tài được nghiên cứu và xây dựng trên nền tảng công nghệ ASP NET CORE 3.1 cùng với cơ sở dữ liệu SQL và hai ngôn ngữ lập trình là C Sharp và Python. Với sự hướng tới là phân tích dữ liệu để có thể ứng dụng tùy vào nhu cầu sử dụng.

1.2. Mục tiêu đề tài

Thời đại ngày nay, với sự phát triển không ngừng của Big Data, việc thu thập và sử dụng dữ liệu đóng vai trò quan trọng trong việc tạo ra giá trị, định hình quyết định và đưa ra các hành động chiến lược.

Từ đó bản thân em đã hướng đến việc nghiên cứu các phương pháp thu thập dữ liệu sau đó phân tích dữ liệu và ứng dụng vào thực tiễn.

1.3. Phạm vi đề tài

Phạm vi được đặt ra là nghiên cứu được phương pháp thu thập, phân tích, quản lý dữ liệu, chi tiết như sau:

- Thu thập được dữ liệu.
- Làm sạch được dữ liệu đã thu thập.
- Phân tích dữ liệu và trực quan ra biểu đồ.
- Cách quản lý dữ liệu lớn.

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

2.1. Lý thuyết

2.1.1. Big Data

Big Data (dữ liệu lớn) là thuật ngữ được sử dụng để miêu tả quy mô và đa dạng của dữ liệu mà các công cụ và phương pháp truyền thống khó có thể xử lý. Big Data được đặc trưng bởi ba yếu tố chính: Volume (khối lượng), Velocity (tốc độ) và Variety (đa dạng). Tuy nhiên, theo thời gian, đã xuất hiện thêm các yếu tố khác như Veracity (độ tin cậy) và Value (giá trị) để đánh giá mức độ quan trọng và chất lượng của dữ liệu.

Các nguồn dữ liệu lớn có thể bao gồm dữ liệu từ máy chủ web, mạng xã hội, cảm biến, thiết bị di động, giao dịch tài chính, dữ liệu y tế và nhiều nguồn khác. Quy mô và phức tạp của Big Data đòi hỏi các công nghệ và phương pháp mới để xử lý, lưu trữ, truy xuất và phân tích dữ liệu.

Big Data thường chứa các loại dữ liệu đa dạng như văn bản, hình ảnh, âm thanh, video, dữ liệu người dùng, dữ liệu máy móc và nhiều hơn nữa. Điều này đòi hỏi các công cụ và phương pháp để hiểu và xử lý các dạng dữ liệu này.

Big Data có tiềm năng mang lại nhiều lợi ích và cơ hội trong nhiều lĩnh vực, bao gồm kinh doanh, khoa học, y tế, tài chính và hơn thế nữa. Các công nghệ và công cụ phổ biến được sử dụng trong Big Data bao gồm:

Xu hướng phát triển của dữ liệu lớn và ứng dụng của nó vào cuộc sống đang ngày càng phát triển và đa dạng. Sự tiến bộ trong công nghệ và phân tích dữ liệu mở ra nhiều cơ hội mới để tận dụng và khai thác dữ liệu lớn để đưa ra quyết định thông minh và cải thiện chất lượng cuộc sống. Xu hướng phát triển hiện tại xoay quanh các khía cạnh sau:

- Tăng cường khả năng xử lý dữ liệu: Với lượng dữ liệu ngày càng tăng, xu hướng phát triển là tăng cường khả năng xử lý dữ liệu bằng cách sử dụng công nghệ mới như xử lý dữ liệu song song, hệ thống phân tán và tính toán đám mây. Điều này giúp giảm thời gian xử lý và tăng hiệu suất.
- Phân tích dữ liệu lớn: Việc phân tích dữ liệu lớn để tìm kiếm thông tin quan trọng, xu hướng và mô hình hóa dữ liệu là một xu hướng quan trọng. Các phương pháp

phân tích dữ liệu lớn bao gồm khai phá dữ liệu, học máy, khai thác dữ liệu và trí tuệ nhân tạo.

- **Trực quan hóa dữ liệu:** Trực quan hóa dữ liệu đóng vai trò quan trọng trong việc hiểu và truyền tải thông tin từ dữ liệu lớn. Công cụ và kỹ thuật trực quan hóa dữ liệu như biểu đồ, bản đồ, biểu đồ tương tác và hình ảnh 3D giúp hiển thị dữ liệu một cách rõ ràng và dễ hiểu.

2.1.2. Xử lý phân tích dữ liệu

Quá trình thu thập, làm sạch và trực quan hóa dữ liệu là quá trình quan trọng trong xử lý và phân tích dữ liệu. Bao gồm các bước sau:

Thu thập dữ liệu: Thu thập thông tin từ nguồn dữ liệu khác nhau như cơ sở dữ liệu, tệp văn bản, trang web, API hoặc cảm biến, thiết bị IoT.

Làm sạch dữ liệu: Xử lý dữ liệu thô để đảm bảo tính chính xác, tin cậy và đồng nhất. Loại bỏ dữ liệu trùng lặp, xử lý dữ liệu thiếu, xử lý ngoại lệ và chuyển đổi dữ liệu vào định dạng phù hợp.

Phân tích dữ liệu: Áp dụng phương pháp thống kê, khai phá dữ liệu và thuật toán học máy để tìm hiểu mẫu, xu hướng và thông tin giá trị từ dữ liệu.

Trực quan hóa dữ liệu: Biểu diễn dữ liệu dưới dạng đồ thị, biểu đồ hoặc hình ảnh để hiển thị một cách trực quan và dễ hiểu.

Lợi ích của quá trình này bao gồm:

- **Hiểu rõ hơn về dữ liệu:** Cung cấp cái nhìn tổng quan về dữ liệu và mối quan hệ giữa các biến.
- **Phát hiện insights và xu hướng:** Phát hiện ra thông tin giá trị, mẫu ẩn và xu hướng trong dữ liệu.
- **Giao tiếp và truyền tải thông tin:** Truyền đạt thông tin phức tạp một cách dễ hiểu và trực quan.
- **Phát triển và cải thiện quy trình:** Xác định điểm yếu, cải thiện hiệu suất và tối ưu hóa quy trình hoạt động hiện tại.
- **Dự đoán và dự báo:** Dự đoán xu hướng, mô hình hóa dữ liệu và đưa ra dự báo cho tương lai.

Tóm lại, quá trình thu thập, làm sạch và trực quan hóa dữ liệu đóng vai trò quan trọng trong khai thác thông tin từ dữ liệu và hỗ trợ quyết định thông minh và phát triển.

2.2. Kỹ thuật

Về mặt kỹ thuật, các công cụ và kỹ thuật sử dụng trong đồ án của em bao gồm:

- Phần mềm: Visual Studio Enterprise 2019, Microsoft SQL Server Management Studio 18, Spyder, Visual Studio Code.
- Công nghệ: Asp.Net Core 3.1
- Mô hình ứng dụng: Mô hình Model –View – Controller.
- Ngôn ngữ lập trình: Ngôn ngữ C#, JavaScript, Python.
- Ngôn ngữ thiết kế giao diện: HTML, CSS, Bootstrap.
- Thư viện: Entity Framework Core, PyScript, Chart.js và các thư viện trong Python.
- Ngoài ra còn sử dụng nhiều công cụ như: Chrome Developer Tools, GitHub,...

2.2.1. Ngôn ngữ C#

2.2.1.1. Khái niệm

C# (C Sharp) là một ngôn ngữ lập trình được phát triển bởi Microsoft và được sử dụng để phát triển các ứng dụng cho nhiều nền tảng, bao gồm Windows, web, cloud, và các thiết bị di động.

C# là một ngôn ngữ lập trình dễ sử dụng với cú pháp gần giống với C++ và Java, cung cấp các tính năng như định nghĩa kiểu dữ liệu, biến, hàm, và các cấu trúc điều khiển. Nó còn cung cấp các tính năng mạnh mẽ như lập trình đối tượng (OOP), tự động quản lý bộ nhớ và hỗ trợ lập trình song song.

2.2.1.2. Ứng dụng

C# có rất nhiều ứng dụng trong các lĩnh vực khác nhau, bao gồm:

Phát triển ứng dụng Windows: C# là một trong những ngôn ngữ chính để phát triển các ứng dụng cho hệ điều hành Windows. Nó cung cấp một số công cụ và thư viện mạnh mẽ để phát triển các ứng dụng Windows Forms và WPF.

Phát triển trang web: C# cũng là một ngôn ngữ phổ biến để phát triển các trang web bằng ASP.NET. ASP.NET cung cấp một khung lập trình để sử dụng và mạnh mẽ để phát triển các trang web dinh dưỡng với tính năng tự động quản lý bộ nhớ và tốc độ nhanh.

2.2.2. Ngôn ngữ Python

Python là một ngôn ngữ lập trình thông dịch và đa năng, nổi tiếng với cú pháp đơn giản và dễ hiểu. Nó được sử dụng rộng rãi trong nhiều lĩnh vực, bao gồm phân tích dữ liệu, máy học, trí tuệ nhân tạo, web development và nhiều lĩnh vực khác.

Dưới đây là một số thư viện phổ biến liên quan đến Python:

- **NumPy:** NumPy là một thư viện toán học cho Python, cung cấp hỗ trợ cho việc làm việc với mảng đa chiều và các phép toán số học trên mảng. Nó là một trong những thư viện cốt lõi cho tính toán khoa học trong Python.
- **Plotly:** Plotly là một thư viện tạo đồ thị và biểu đồ tương tác trong Python. Nó cung cấp các công cụ mạnh mẽ để tạo ra đồ thị trực quan, biểu đồ thống kê và biểu đồ khoa học dễ dàng.
- **Pandas:** Pandas là một thư viện dữ liệu mạnh mẽ cho Python. Nó cung cấp cấu trúc dữ liệu linh hoạt và công cụ phân tích dữ liệu để làm việc với dữ liệu có cấu trúc và không có cấu trúc. Pandas hỗ trợ nhiều loại dữ liệu, bao gồm bảng dữ liệu (DataFrame) và chuỗi dữ liệu (Series).
- **DateTime:** DateTime là một module trong Python cung cấp các lớp và hàm để làm việc với thời gian và ngày tháng. Nó cho phép bạn tạo, xử lý và định dạng các đối tượng thời gian và ngày tháng.
- **Dateutil:** Dateutil là một thư viện Python mở rộng cho phép xử lý dữ liệu thời gian và ngày tháng phức tạp. Nó cung cấp các phương pháp linh hoạt để phân tích, chuyển đổi và tính toán với các đối tượng thời gian và ngày tháng.
- Các thư viện trên đều rất phổ biến và hữu ích trong việc phân tích dữ liệu, trực quan hóa và xử lý thời gian trong Python.

2.2.3. Mô hình MVC

MVC là viết tắt của Model-View-Controller, đó là một kiểu mô hình lập trình phổ biến được sử dụng trong việc phát triển các ứng dụng web và di động.

Cách hoạt động của MVC như sau:

- **Model:** Là một lớp chứa dữ liệu và các thuật toán xử lý dữ liệu. Model có thể tương tác với cơ sở dữ liệu để lấy và cập nhật dữ liệu.

- View: Là một lớp chứa các giao diện người dùng và các thao tác liên quan đến giao diện người dùng. View tạo ra giao diện để người dùng có thể tương tác với ứng dụng.
- Controller: Là một lớp chứa các xử lý sự kiện và các thao tác liên quan đến sự kiện. Controller xử lý các yêu cầu từ người dùng và gửi các yêu cầu đến Model hoặc View.

Trong MVC, Model, View, và Controller là ba thành phần tách biệt và không giao tiếp trực tiếp với nhau. MVC giúp tách rời các thành phần của ứng dụng, giúp cho việc phát triển và bảo trì trở nên dễ dàng hơn.

2.2.4. Spyder và Visual Studio Code

Spyder và Visual Studio Code (VS Code) đều là môi trường phát triển tích hợp (IDE) cho Python và hỗ trợ nhiều tính năng liên quan đến trực quan hóa và biểu đồ. Tuy nhiên, có một số khác biệt giữa hai công cụ này:

Spyder: Spyder là một IDE được thiết kế đặc biệt cho việc phân tích dữ liệu và khoa học dữ liệu trong Python. Nó cung cấp một giao diện người dùng để sử dụng với các công cụ phân tích dữ liệu tích hợp như IPython console, trình soạn thảo mã, trình duyệt biến số và bảng điều khiển trực quan. Spyder có sẵn các thư viện phổ biến như NumPy, Pandas và Matplotlib, giúp người dùng dễ dàng thực hiện các tác vụ trực quan hóa dữ liệu và biểu đồ.

Visual Studio Code: VS Code là một IDE linh hoạt và mở rộng được sử dụng rộng rãi cho nhiều ngôn ngữ lập trình, bao gồm Python. Mặc dù không chuyên dụng cho khoa học dữ liệu nhưng VS Code cung cấp một loạt các tiện ích và tiện ích mở rộng hỗ trợ trực quan hóa dữ liệu và biểu đồ. Các tiện ích như Jupyter Notebook, Python Interactive Window và các tiện ích mở rộng như Plotly và Pandas giúp người dùng tạo và hiển thị biểu đồ dễ dàng trong VS Code.

Cả Spyder và Visual Studio Code đều hỗ trợ việc trực quan hóa dữ liệu và biểu đồ thông qua các thư viện như Matplotlib, Plotly và Pandas. Cả hai công cụ đều có tính năng gỡ lỗi, hiển thị biến số và cho phép thực thi từng dòng mã một. Sự lựa chọn giữa Spyder và VS Code phụ thuộc vào sở thích cá nhân, yêu cầu công việc và môi trường phát triển của người dùng.

2.2.5. SQL Server

Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ được phát triển bởi Microsoft. Là một máy chủ cơ sở dữ liệu, nó là một sản phẩm phần mềm có chức năng chính là lưu trữ và truy xuất dữ liệu theo yêu cầu của các ứng dụng phần mềm khác. Có thể chạy trên cùng một máy tính hoặc trên một máy tính khác trên mạng (bao gồm cả Internet).

SQL Server được xây dựng dựa trên SQL, được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn lên đến Tera – Byte cùng lúc phục vụ cho hàng ngàn user. SQL Server cung cấp đầy đủ các công cụ cho việc quản lý từ nhận diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL.

Tính di động: SQL có thể được sử dụng trong chương trình trong PCs, servers, laptops, và thậm chí cả mobile phones.

Ngôn ngữ tương tác: Language này có thể được sử dụng để giao tiếp với cơ sở dữ liệu và nhận câu trả lời cho các câu hỏi phức tạp trong vài giây.

Multiple data views: Với sự trợ giúp của ngôn ngữ SQL, người dùng có thể tạo các hiển thị khác nhau về cấu trúc cơ sở dữ liệu và cơ sở dữ liệu cho những người dùng khác nhau.

2.2.6. ASP.NET Core 3.1 và Entity Framework Core

ASP.NET Core 3.1 là một framework phát triển ứng dụng web mạnh mẽ và đa nền tảng. Nó cung cấp các tính năng như routing, middleware, dependency injection và hỗ trợ RESTful API. Với khả năng mở rộng và tương thích tốt với các dịch vụ đám mây, ASP.NET Core 3.1 là lựa chọn phổ biến cho việc xây dựng ứng dụng web liên quan đến Big Data.

Entity Framework Core là một ORM (Object-Relational Mapping) framework trong .NET, cung cấp một cách tiếp cận trừu tượng hóa việc làm việc với cơ sở dữ liệu. Nó giúp giảm thiểu công việc lập trình tương tác với cơ sở dữ liệu và cung cấp tính năng như khả năng thay đổi cấu trúc cơ sở dữ liệu và truy vấn dữ liệu dễ dàng. Entity Framework Core cũng hỗ trợ các dịch vụ lưu trữ Big Data như Microsoft Azure Cosmos DB. Khi làm việc với Big Data, ASP.NET Core 3.1 và Entity Framework Core có thể tương tác với các dịch vụ và công nghệ Big Data như Hadoop, Spark, Cassandra, hoặc MongoDB. Chúng có thể được sử dụng để lưu trữ và truy vấn dữ liệu từ các nguồn dữ liệu lớn, thực hiện phân

tích và truy vấn dữ liệu phức tạp, và xử lý dữ liệu trực quan thông qua các công cụ trực quan hóa dữ liệu như biểu đồ và bản đồ.

Tóm lại, ASP.NET Core 3.1 và Entity Framework Core là những công nghệ mạnh mẽ trong việc phát triển ứng dụng web và có thể được kết hợp với Big Data để làm việc với dữ liệu lớn và phức tạp

2.2.7. Chart.js

2.2.7.1. Khái niệm

Chart.js là một thư viện JavaScript mã nguồn mở được sử dụng để tạo và hiển thị các biểu đồ trực quan trên trang web. Nó cung cấp một cách đơn giản và mạnh mẽ để tạo ra các biểu đồ dựa trên dữ liệu số liệu và hiển thị chúng dưới dạng đồ thị thanh, đồ thị tròn, đồ thị đường, biểu đồ phức tạp hơn và nhiều loại biểu đồ khác

2.2.7.2. Ưu điểm

Dễ sử dụng: Chart.js có cú pháp đơn giản và dễ hiểu, giúp người dùng dễ dàng tạo và tùy chỉnh các biểu đồ trực quan theo ý muốn.

Đa dạng loại biểu đồ: Thư viện hỗ trợ nhiều loại biểu đồ phổ biến như đồ thị thanh, đồ thị tròn, đồ thị đường, biểu đồ vùng, biểu đồ hình cột và biểu đồ radar.

Linh hoạt và tùy chỉnh: Chart.js cho phép người dùng tùy chỉnh các thuộc tính, màu sắc, phong cách và hiệu ứng của biểu đồ để phù hợp với giao diện và yêu cầu cụ thể.

Responsive và tương thích di động: Các biểu đồ tạo bằng Chart.js có khả năng thích ứng với kích thước màn hình và tương thích trên các thiết bị di động, giúp hiển thị dữ liệu một cách tốt nhất trên mọi nền tảng.

Hỗ trợ giao diện tương tác: Người dùng có thể tương tác với các biểu đồ bằng cách di chuột, chạm hoặc bấm vào các phần tử của biểu đồ để xem thông tin chi tiết và tương tác với dữ liệu.

Tích hợp dễ dàng: Chart.js có thể được tích hợp vào các dự án web hiện có một cách dễ dàng thông qua mã HTML và JavaScript.

Hỗ trợ tài liệu phong phú: Chart.js cung cấp tài liệu chi tiết và ví dụ đa dạng để hướng dẫn người dùng sử dụng và tùy chỉnh các biểu đồ.

2.2.8. Pyscript

Pyscript là một thư viện Python được sử dụng để thực hiện các tác vụ liên quan đến việc tạo và thao tác các tệp mã Python. Nó cung cấp các công cụ để tạo ra các tệp mã Python từ các tệp mã nguồn khác nhau và thực thi chúng. Pyscript giúp đơn giản hóa việc tạo ra mã Python và thực thi nó trong quy trình phát triển và tự động hóa công việc.

Một số tính năng chính của Pyscript bao gồm:

Tạo tệp mã Python: Pyscript cho phép tạo ra các tệp mã Python từ các tệp nguồn khác nhau như tệp văn bản, tệp Excel hoặc tệp JSON. Bằng cách sử dụng cú pháp đơn giản, bạn có thể chuyển đổi các tệp dữ liệu này thành mã Python để thực thi hoặc sử dụng trong dự án của mình.

Thực thi mã Python: Pyscript cho phép bạn thực thi các tệp mã Python một cách tự động và linh hoạt. Bạn có thể chạy các tệp mã Python trong một môi trường được định nghĩa trước, đảm bảo các yêu cầu phụ thuộc đúng được cài đặt, hoặc có thể chỉ định các tham số và tùy chọn khác để điều khiển quá trình thực thi.

Tích hợp với công cụ phát triển: Pyscript tích hợp tốt với các công cụ phát triển Python khác như trình biên tập mã nguồn và môi trường phát triển tích hợp (IDE). Điều này cho phép bạn làm việc với Pyscript trong môi trường đã quen thuộc và tận dụng các tính năng bổ sung của các công cụ này để làm việc hiệu quả hơn.

2.2.9. Git và GitHub

Git là một hệ thống quản lý mã nguồn phân tán (Distributed Version Control System) được sử dụng để quản lý mã nguồn và tạo nhánh (branch) trong quá trình phát triển phần mềm. Nó giúp cho lập trình viên giữ những thay đổi trong mã nguồn và dễ dàng quản lý các phiên bản của mã nguồn một cách dễ dàng.

GitHub là một nền tảng lưu trữ dựa trên web dành cho các kho kiểm soát phiên bản. Nó cung cấp một nơi tập trung để lưu trữ và quản lý kho Git, giúp các nhà phát triển dễ dàng cộng tác trong một dự án. GitHub cung cấp các tính năng như theo dõi lỗi, quản lý dự án và các công cụ cộng tác nhóm khiến nó trở thành một nền tảng phổ biến để phát triển phần mềm. Ngoài ra, GitHub cung cấp một nền tảng cho các dự án nguồn mở, giúp các nhà phát triển đóng góp và sử dụng phần mềm nguồn mở dễ dàng hơn.

CHƯƠNG III: CÀI ĐẶT THỬ NGHIỆM

3.1. Phương pháp nghiên cứu

Phương pháp nghiên cứu cho đề tài "Nghiên cứu thu thập, phân tích dữ liệu và ứng dụng thực tiễn" có thể tuân theo các bước sau:

Xác định mục tiêu nghiên cứu: Đầu tiên, xác định rõ mục tiêu của nghiên cứu. Điều này có thể bao gồm việc đặt ra các câu hỏi nghiên cứu cụ thể, mục tiêu dự kiến và kỳ vọng về kết quả của đề tài.

Thu thập dữ liệu: Xác định các nguồn dữ liệu phù hợp để thu thập thông tin cho nghiên cứu. Điều này có thể bao gồm việc tìm hiểu các nguồn dữ liệu có sẵn, sử dụng phương pháp khảo sát, phỏng vấn, hoặc thu thập dữ liệu từ các nguồn trực tuyến.

Là sạch dữ liệu: Sau khi thu thập dữ liệu, tiến hành làm sạch và xử lý dữ liệu. Bước này bao gồm loại bỏ dữ liệu không hợp lệ, điền các giá trị bị thiếu, xử lý ngoại lệ và chuyển đổi dữ liệu vào định dạng phù hợp cho phân tích tiếp theo.

Phân tích dữ liệu: Áp dụng các phương pháp phân tích dữ liệu phù hợp để khám phá mẫu, xu hướng và thông tin hữu ích từ dữ liệu. Điều này có thể bao gồm việc sử dụng các phương pháp thống kê, khai phá dữ liệu và các thuật toán học máy để tìm ra insights từ dữ liệu.

Đánh giá và ứng dụng thực tiễn: Đánh giá kết quả của quá trình phân tích dữ liệu và xác định những ứng dụng thực tiễn của các kết quả nghiên cứu. Điều này có thể bao gồm việc so sánh kết quả với mục tiêu nghiên cứu ban đầu và đánh giá khả năng áp dụng kết quả vào các vấn đề thực tế.

Tổng kết và trình bày kết quả: Cuối cùng, tổng kết kết quả nghiên cứu và trình bày trong báo cáo hoặc bài viết khoa học. Báo cáo nên bao gồm mô tả chi tiết về phương pháp nghiên cứu, kết quả phân tích dữ liệu, và nhận xét về ứng dụng thực tiễn của nghiên cứu.

Quá trình nghiên cứu này yêu cầu sự kiên nhẫn, cẩn thận và kỹ năng trong việc xử lý và phân tích dữ liệu. Đồng thời, việc áp dụng những kiến thức và kỹ năng phù hợp trong lĩnh vực dữ liệu và ứng dụng thực tiễn cũng rất quan trọng

3.2. Phương pháp thực hiện

3.2.1 Thu thập dữ liệu

- Cài đặt thư viện

```
import requests
from bs4 import BeautifulSoup
import json
from urllib.parse import urlparse
import re
```

Lần lượt:

- Được sử dụng để gửi yêu cầu HTTP và nhận phản hồi từ các trang web.
- Được sử dụng để phân tích cú pháp HTML và trích xuất thông tin từ trang web.
- Dùng để làm việc với định dạng JSON.
- Được sử dụng để phân tích URL và trích xuất các thành phần của nó.
- Được sử dụng để thực hiện các phép so khớp mẫu (regular expressions).

```
URL = "https://katjewelry.vn/new-collection"
parsed_url = urlparse(URL)
domain = parsed_url.netloc.split('.')[0]

r = requests.get(URL)
soup = BeautifulSoup(r.content, 'html.parser')
quotes=[] # a list to store quotes

netloc = parsed_url.netloc;
```

Truyền vào URL muốn lấy, sau đó sử dụng “urlparse” để phân tích URL và trích xuất các thành phần của nó. Cụ thể, sau khi đã sử dụng urlparse để phân tích URL thành các thành phần (như scheme, netloc, path, v.v.), netloc chứa phần tên miền của URL. Với dòng tiếp theo lấy phần tên miền của URL từ thuộc tính “netloc” của đối tượng parsed_url, dòng này sẽ trả về “katjewelry”.

```
table = soup.find('section', attrs = {'class': 'products-view products-view-grid test1'})
for row in table.findAll('div', attrs = {'class', 'col-xs-6 col-sm-4 col-md-4 col-lg-3'}):
    quote = {}
    quote['ProductLink'] = netloc + row.a['href']
    quote['ProductName'] = row.find('a', attrs = {'class', 'line-clamp'}).text.strip().replace('\u00a0', '').replace('\u20ab', '').replace('.', '')
    quote['ProductPrice'] = row.find('span', attrs = {'class', 'price product-price'}).text.replace('\u00a0', '').replace('\u20ab', '').replace(',', '')
    quote['ProductImg'] = row.find('img')['data-lazyload']
    quotes.append(quote)
```

Đoạn này sẽ dựa vào cấu trúc HTML của URL đã nhập để tìm phần tử để xác định vị trí trích xuất thông tin. Với đoạn mã xử lý trên, luồng đi sẽ là tìm và gán phần tử <section> có các lớp được chỉ định A. Tiếp theo sẽ lặp qua từng phần tử <div> có các thuộc tính chỉ định B thuộc phần “table”. Và khởi tạo mảng quote rỗng.

Tại đây, quote sẽ tìm và lưu trữ các thông tin sản phẩm. Bao gồm:

- Lấy liên kết sản phẩm bằng cách ghép netloc (tên miền) với giá trị thuộc tính href của thẻ <a> trong phần tử row hiện tại.
- Tìm thẻ <a> có thuộc tính lớp chỉ định trong phần tử row hiện tại và trích xuất tên sản phẩm bằng cách loại bỏ các ký tự không mong muốn.
- Tìm thẻ có thuộc tính lớp chỉ định trong phần tử row hiện tại và trích xuất giá sản phẩm bằng cách loại bỏ các ký tự không mong muốn.
- Tìm thẻ trong phần tử row hiện tại và lấy giá trị của thuộc tính data-lazyload, đại diện cho hình ảnh sản phẩm.
- Cuối cùng sẽ thêm quote vào mảng quotes.

```
filename = domain + '.json'
with open(filename, 'w') as f:
    json.dump(quotes, f)
```

Sau khi hoàn thành vòng lặp sẽ tiến hành lưu lại với tên là domain và đuôi là .json. Với mở rộng là tạo một tệp mới hoặc ghi đè lên tệp hiện có nếu nó đã tồn tại. Ghi dữ liệu của quotes vào tệp theo định dạng JSON.

3.2.2 Làm sạch dữ liệu

- Dữ liệu ban đầu**

Bảng 3. 1 Dữ liệu ban đầu

ID	Name	Birthday	Phone	Address
1	mIChAel mIcHALek#	08/07/19	84333605993	123 Nguyễn Thị Minh Khai, P. Bến Thành, Q.1, TP. HCM
2	Andrew \$Jimenez	09/10/18	0973444062	456 Lê Lợi, P. Bến Nghé, Q.1, TP.HCM
3	Ann Gow	30061990	0338262954	789 Lê Duẩn, P. Bến Thành, Q.1, TP.HCM
4	James Chen	9/6/2001 12:00:00	84966068026	2122 Nguyễn Hữu Cánh, P. 22, Q. Bình Thạnh, TP.HCM
5	Dollie Martinez	19970812	84767065885	1920 Trần Quang Khải, P. Tân Định, Q.1, TP.HCM
6	Roger Callender	1998-06-02 00:00:00	84339769174	1618 Võ Văn Kiệt, P. Cầu Kho, Q.1, TP. HCM
7	David Liff	01/05/n20	+84349251856	1415 Nguyễn Văn Linh, P. Tân Phong, Q.7, TP.HCM
8	RubEN rAy	20011128	84964118798	1012 Điện Biên Phủ, P. 25, Q.Bình Thạnh, TP.HCM
9	JaMeS dEmERs&	2001-12-01 00:00:00	0782904001	2930 Phạm Ngọc Thạch, P. 6, Q.3, TP.HCM
10	anDrEa andreWS!	15/01/98	+84862606446	2728 Nguyễn Thái Bình, P. Nguyễn Thái Bình, Q.1, TP.HCM
10	Otis Arnold@	10/07/90	84973180839	2526 Phan Văn Trị, P. 10, Q.Gò Vấp, TP.HCM
11	Elmer Feezell	1998-07-03 00:00:00.12	+84967617424	2324 Cách Mạng Tháng Tám, P. 12, Q.10, TP.HCM
12	;Kim Biddle	1999-09-15 00:00:00	84971262200	3738 Phan Văn Hân, P. Tân Sơn Nhì, Q.Tân Phú, TP.HCM
13	Paul Ruper	Jun-03-2000	84356173988	3536 Huỳnh Tịnh Của, P. Tân Thành, Q.Tân Phú, TP.HCM
14	Robin Ledford	19980418	84126737070	3334 Hoàng Sa, P. Tân Định, Q.1, TP.HCM
15	Elmer Feezell	1996-03-10	0971959761	3132 Trương Định, P. Bến Thành, Q.1, TP.HCM
16	eIMER mILLER	10-04-2001	0867005738	5254 Nguyễn Trọng Tuyển, P. 8, Q.Phú Nhuận, TP.HCM
17	BiabEN rAy	19/08/1995	128320706	4850 Cao Thắng, P. 5, Q.3, TP.HCM
18	Ruper Otis	980629	0769789386	
14	Robin Ledford	20180418	84126737073	4042 Nguyễn Văn Đậu, P. 5, Q.Bình Thạnh, TP.HCM

- Cài đặt thư viện**

```
import pandas as pd
from dateutil.parser import parse
import datetime
```

Đọc dữ liệu:

```
df = pd.read_excel('D:/lby/Nam4/HKII/BIGDATA/messy.xlsx', sheet_name='Sheet1')
```


- **Làm sạch dữ liệu**

Các bước làm sạch bao gồm:

- #a. Loại bỏ tất cả cột rỗng, đổi tên cột dữ liệu
- #b. Định dạng format chữ theo kiểu title – loại bỏ ký tự đặc biệt
- #c. Loại bỏ khoảng trắng thừa
- #d. Định dạng lại theo vùng 84
- #e. Định dạng ngày sinh về kiểu yyyy-MM-dd ('%Y-%m-%d')
- #f. Loại bỏ trùng ID -> Lưu trữ lại sang một file khác.
- #g. Lọc dữ liệu
- #h. Xuất file excel

a. Loại bỏ tất cả cột rỗng, đổi tên cột dữ liệu

```
df.columns
df = df.dropna(axis=1, how='all')
df.columns = ['staff_id', 'staff_name', 'staff_birthday',
'staff_phone', 'staff_address']
```

b. Định dạng format chữ theo kiểu title – loại bỏ ký tự đặc biệt

```
df['staff_name'] = df['staff_name'].str.replace(r"^[^\\w\\s]", "")
df['staff_name'] = df['staff_name'].str.title()
```

Với: ^ đảo - khớp với \\w các kí tự là chữ số và \\s không phải khoảng trắng

c. Loại bỏ khoảng trắng thừa

```
#thay thế giá trị NaN thành "trống"
df['staff_address'] = df['staff_address'].fillna('trống')
#loại bỏ khoảng trắng thừa ở đầu cuối
df['staff_address'] = df['staff_address'].str.strip()
#loại bỏ khoảng trắng P. với từ tiếp theo, sau đó tách chuỗi thành danh sách các
từ và ghép lại chúng thành một chuỗi mới chỉ với một khoảng trắng duy nhất giữa
các từ
df['staff_address'] = df['staff_address'].str.replace(r'(P\\.|Q\\.|TP/\\.\\s+',
r'\\1').replace(r'\\s+', ' ', regex=True)
```

d. Định dạng lại theo vùng 84

```
df['staff_phone']=df['staff_phone'].astype(str)
df['staff_phone'] = df['staff_phone'].apply(lambda x: x if x.startswith('84')
else "84"+x)
```

e. Định dạng ngày sinh về kiểu yyyy-MM-dd ('%Y-%m-%d')

```
for index, bd in df['staff_birthday'].items():
    try:
        date = parse(bd)
        date_obj = parse(date)
        df.loc[index, 'staff_birthday'] = date_obj.strftime('%Y-%m-%d')

    except:
        date_formats = ["%d/%m/%y", "%y%m%d", "%d/%m/%Y", "%d-%m-%Y", "%Y%m%d",
                        "%d/%m/%y", "%b-%d-%Y", "%Y/%m/%d", "%d%m%Y", "%m/%d/%Y %H:%M:%S", "%Y-%m-%d",
                        "%H:%M:%S", "%Y-%m-%d %H:%M:%S.%f"]
        for date_format in date_formats:
            try:
                dt = datetime.datetime.strptime(bd, date_format)
                df.loc[index, 'staff_birthday'] = dt.strftime("%Y-%m-%d")
                break
            except ValueError:
                pass
```

Tại đây bắt các định dạng thời gian ngoại lệ không có trong thư viện datetime. Có một ngược điểm là phải thay đổi nếu có ngoại lệ khác không thuộc.

f. Loại bỏ trùng ID -> Lưu trữ lại sang một file khác

```
#tìm những staff id trùng vào một df
df_idDup = df[df['staff_id'].duplicated()]
#xóa trùng
df = df.drop_duplicates("staff_id", keep='first')
```

g. Lọc dữ liệu

```
#####Lọc theo ngày sinh (trước năm 2000)
#Chuyển đổi kiểu chuỗi thành đối tượng datetime.date
df['staff_birthday'] = pd.to_datetime(df['staff_birthday']).dt.date
df_date2000 = df[df.staff_birthday <
datetime.date(2000,1,1)].reset_index(drop=True)

#####Lọc theo tuổi (>=22)
#365 + 1/4 - 1/100 + 1/400 = 365.2425
today = datetime.date.today()
age = (today - df['staff_birthday']) // datetime.timedelta(days=365.25)
df_age22 = df[age >= 22].reset_index(drop=True)
df_age22 = df_age22.sort_values(by=['staff_birthday'])

#####lọc address
df_tanphu = df.loc[df['staff_address'].str.contains('Q\.\.Tân Phú')]
```

h. Xuất file excel

```
#lưu df_idDup và df vào chung 1 excel
with pd.ExcelWriter('datacleaning.xlsx') as writer:
    df.to_excel(writer, sheet_name='staff', index=False)
    df_idDup.to_excel(writer, sheet_name='staffDup', index=False)
```

- **Kết quả**

- a. Đã làm sạch

Bảng 3. 2 Dữ liệu đã sạch

Staff_id	staff_name	staff_birthday	staff_phone	staff_address
1	Michael Michalek	2019-07-08	84333605993	123 Nguyễn Thị Minh Khai, P.Bến Thành, Q.1, TP.HCM
2	Andrew Jimenez	2018-10-09	84973444062	456 Lê Lợi, P.Bến Nghé, Q.1, TP.HCM
3	Ann Gow	1990-06-30	84338262954	789 Lê Duẩn, P.Bến Thành, Q.1, TP.HCM
4	James Chen	2001-09-06	84966068026	2122 Nguyễn Hữu Cảnh, P.22, Q.Bình Thạnh, TP.HCM
5	Dollie Martinez	1997-08-12	84767065885	1920 Trần Quang Khải, P.Tân Định, Q.1, TP.HCM
6	Roger Callender	1998-06-02	84339769174	1618 Võ Văn Kiệt, P.Cầu Kho, Q.1, TP.HCM
7	David Liff	2020-05-01	84349251856	1415 Nguyễn Văn Linh, P.Tân Phong, Q.7, TP.HCM
8	Ruben Ray	2001-11-28	84964118798	1012 Điện Biên Phủ, P.25, Q.Bình Thạnh, TP.HCM
9	James Demers	2001-12-01	84782904001	2930 Phạm Ngọc Thạch, P.6, Q.3, TP.HCM
10	Andrea Andrews	1998-01-15	84862606446	2728 Nguyễn Thái Bình, P.Nguyễn Thái Bình, Q.1, TP.HCM
11	Elmer Feezell	1998-07-03	84967617424	2324 Cách Mạng Tháng Tám, P.12, Q.10, TP.HCM
12	Kim Biddle	1999-09-15	84971262200	3738 Phan Văn Hân, P.Tân Sơn Nhì, Q.Tân Phú, TP.HCM
13	Paul Ruper	2000-06-03	84356173988	3536 Huỳnh Tịnh Của, P.Tân Thành, Q.Tân Phú, TP.HCM
14	Robin Ledford	1998-04-18	84126737070	3334 Hoàng Sa, P.Tân Định, Q.1, TP.HCM
15	Elmer Feezell	1996-03-10	84971959761	3132 Trương Định, P.Bến Thành, Q.1, TP.HCM
16	Elmer Miller	2001-04-10	84867005738	5254 Nguyễn Trọng Tuyển, P.8, Q.Phú Nhuận, TP.HCM
17	Biaben Ray	1995-08-19	84128320706	4850 Cao Thắng, P.5, Q.3, TP.HCM
18	Ruper Otis	1998-06-29	84769789386	trống

b. Dữ liệu trùng

Bảng 3. 3 Dữ liệu trùng

Staff_id	staff_name	staff_birthday	staff_phone	staff_address
10	Otis Arnold	1990-07-10	84973180839	2526 Phan Văn Trị, P.10, Q.Gò Vấp, TP.HCM
14	Robin Ledford	2018-04-18	84126737073	4042 Nguyễn Văn Đậu, P.5, Q.Bình Thạnh, TP.HCM

3.2.3 Trục quan dữ liệu

3.2.3.1 Trục quan với python

- Cài đặt thư viện

```
import pandas as pd
import plotly.express as px
```

Lần lượt:

- Sử dụng để làm việc với dữ liệu và tạo các bảng dữ liệu.
- Thư viện để tạo ra các biểu đồ và biểu đồ tương tác.

- Thực hiện

```
# Đọc dữ liệu từ file JSON
data=pd.read_json("D:/lby/Nam4/HKII/BIGDATA/CuoiKy/hanghoa.json")

df_grouped = data.groupby(['TenHH'])['TriGiaTon'].sum().reset_index()

fig = px.bar(df_grouped, x='TenHH', y='TriGiaTon', title='Top 5 hàng hóa có trị
giá tồn cao nhất',color='TenHH',text=data['TongTon'])
fig.update_traces(hovertemplate='Hàng hóa: %{x}<br>số lượng: %{y:,.0f}VNĐ')

fig.show()
```

Lần lượt:

- Đọc dữ liệu từ file hanghoa.json.
- Gom nhóm dataframe df_grouped gồm cột TenHH và TongTon.

- Tạo biểu đồ cột với dữ liệu đầu vào là df_grouped, cột x là TenHH, cột y là TongTon. Đặt tên tiêu đề của biểu đồ tại title, đổi màu các cột với color và hiển thị giá trị TriGiaTon trên mỗi cột.
- Cập nhật nội dung khi di chuột vào sẽ hiển thị thông tin.
- Cuối cùng hiển thị biểu đồ.

• Kết quả

Có thể tương tác với biểu đồ



Hình 3. 1 Trực quan với python

3.2.3.2 Trục quan với ChartJs

- Lấy dữ liệu truyền vào view → Lấy được dữ liệu gồm top 5 hàng hóa có trị giá tồn kho cao nhất.

```
public IActionResult TrucQuan()
{
    var results = context.TonKho
        .Join(context.ChiTietPhieuNhap, tk => tk.Idctpn, ctn => ctn.Id, (tk,
        ctn) => new { tk, ctn })
        .Join(context.HangHoa, x => x.ctn.Idhh, hh => hh.Id, (x, hh) => new {
        x.tk, x.ctn, hh })
        .GroupBy(x => new { x.hh.MaHh, x.hh.TenHh })
        .Select(g => new TonKhoModel
        {
            MaHH = g.Key.MaHh,
            TenHH = g.Key.TenHh,
            SL = (double)g.Sum(tk => tk.tk.SoLuong),
            Gia = (double)g.Sum(x => x.tk.SoLuong * (x.ctn.Price * (1 +
            x.ctn.Thue / 100)))
        })
        .OrderByDescending(r => r.Gia)
        .Take(5)
        .ToList();

    ViewBag.Results = results.OrderBy(r => r.Gia);
    return View();
}
```

- Khởi tạo HTML

```
<div class="col-xl-8 col-lg-7">
  <div class="card shadow mb-4">
    <div class="card-header py-3">
      <h6 class="m-0 font-weight-bold text-primary">
        Top 5 hàng hóa có trị giá tồn kho lớn nhất
      </h6>
    </div>
    <div class="card-body">
      <div class="chart-bar">
        <canvas id="myBarChart"></canvas>
      </div>
    </div>
  </div>
</div>
```

Khởi tạo HTML với một thẻ canvas có id “myBarChart”. Thẻ canvas được sử dụng để vẽ biểu đồ cột bằng JavaScript hoặc thư viện biểu đồ như Chart.js.

- Truyền dữ liệu vào Chart

- Tạo 2 mảng chứa danh sách giá trị TenHH và Gia. Khởi tạo datasets cho Chart.

```
// Tạo mảng các giá trị Gia từ dữ liệu results
var giaValues = [];
var labels = [];
var colors = ["#4e73df", "#1cc88a", "#36b9cc", "#f6c23e", "#e74a3b"]; // Mảng các màu

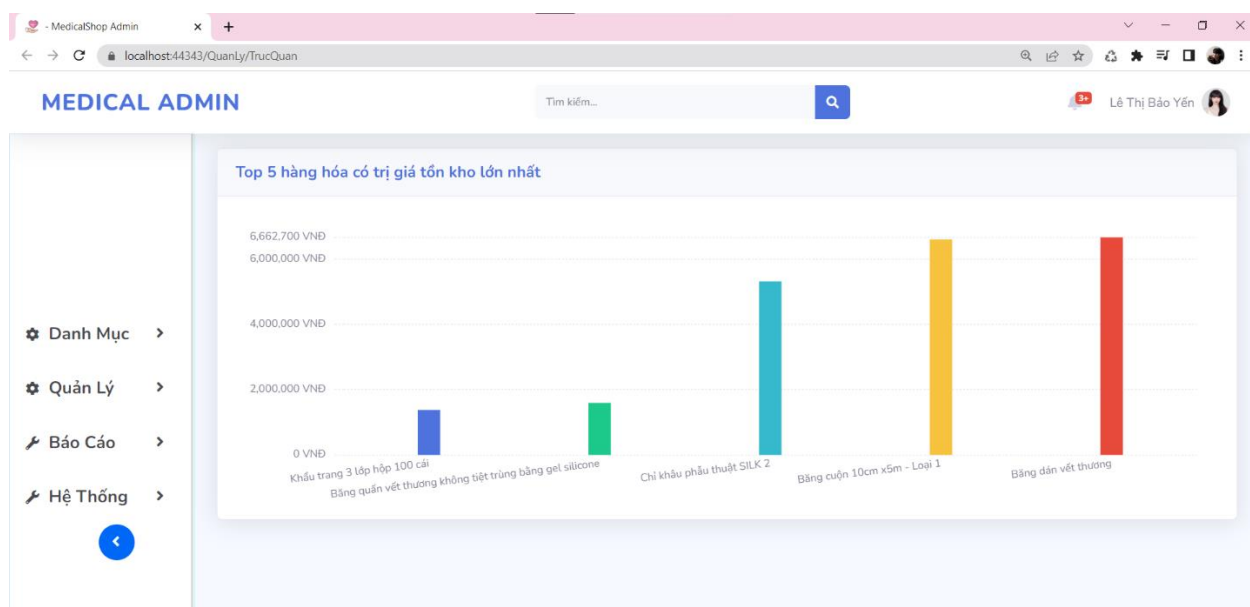
var datasets = [{
  label: "Giá trị tồn",
  backgroundColor: colors, // Gán mảng màu cho backgroundColor
  hoverBackgroundColor: colors, // Gán mảng màu cho hoverBackgroundColor
  borderColor: colors, // Gán mảng màu cho borderColor
  data: giaValues,
}];

// Lặp qua dữ liệu results và trích xuất giá trị Gia và nhãn TenHH
@foreach (var item in ViewBag.Results)
{
  @:giaValues.push(@item.Gia);
  @:labels.push("@Html.Raw(item.TenHH)");
}
```


- Truyền labels và datasets vào Chart, format hiển thị lại với kiểu tiền VNĐ.

```
var ctx = document.getElementById("myBarChart");
var myBarChart = new Chart(ctx, {
  type: 'bar',
  data: {
    labels: labels,
    datasets: datasets,
  },
  ...
  scales: {
    ...
    yAxes: [{
      ticks: {
        ...
        callback: function (value, index, values) {
          return number_format(value) + ' VNĐ';
        }
      },
      ...
    }],
  },
  ...
  tooltips: {
    ...
    callbacks: {
      label: function (tooltipItem, chart) {
        var datasetLabel = chart.datasets[tooltipItem.datasetIndex].label || '';
        return datasetLabel + ': ' + number_format(tooltipItem.yLabel) + ' VNĐ' ;
      }
    }
  },
  ...
});
```

- Kết quả



Hình 3. 2 Trực quan với ChartJs

3.2.3.3 Trục quan với Pyscript

- Lấy dữ liệu truyền vào view → Lấy được dữ liệu gồm top 5 hàng hóa có trị giá tồn kho cao nhất.

```
public IActionResult TestTQ()
{
    var results = context.TonKho
        .Join(context.ChiTietPhieuNhap, tk => tk.Idctpn, ctn => ctn.Id, (tk,
ctn) => new { tk, ctn })
        .Join(context.HangHoa, x => x.ctn.Idhh, hh => hh.Id, (x, hh) => new {
x.tk, x.ctn, hh })
        .GroupBy(x => new { x.hh.MaHh, x.hh.TenHh })
        .Select(g => new TonKhoModel
        {
            MaHH = g.Key.MaHh,
            TenHH = g.Key.TenHh,
            SL = (double)g.Sum(tk => tk.tk.SoLuong),
            Gia = (double)g.Sum(x => x.tk.SoLuong * (x.ctn.Price * (1 +
x.ctn.Thue / 100)))
        })
        .OrderByDescending(r => r.Gia)
        .Take(5)
        .ToList();

    ViewBag.Results = results.OrderBy(r => r.Gia);
    return View();
}
```

- Khởi tạo HTML
- Cần nhúng link stylesheet và script sau

```
<link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
<script defer src="https://pyscript.net/alpha/pyscript.js"></script>
```

- Cài đặt 2 gói sau cho môi trường python

```
<py-config>
  packages = ["matplotlib", "pandas"]
</py-config>
```

- Import thư viện

```
import pandas as pd
import matplotlib.pyplot as plt
```

Lần lượt:

- Sử dụng để làm việc với dữ liệu và tạo các bảng dữ liệu.
- Để trục quan hóa dữ liệu bằng các biểu đồ.

- Truyền dữ liệu và khởi tạo biểu đồ
- Chuyển đổi dữ liệu truyền từ controller sang dạng Json.

```
var results = ViewBag.Results;
var jsonData = @Html.Raw(Json.Serialize(ViewBag.Results));
```

- Truyền dữ liệu ở dạng Json vào data, khởi tạo biểu đồ cột với cột x,y lần lượt là Tên hàng hóa và tổng số lượng tồn. Trên mỗi cột sẽ hiển thị trị giá tồn của hàng hóa đó.

<py-script>

```
import pandas as pd
import matplotlib.pyplot as plt

data = @jsonData;

def plot(data):

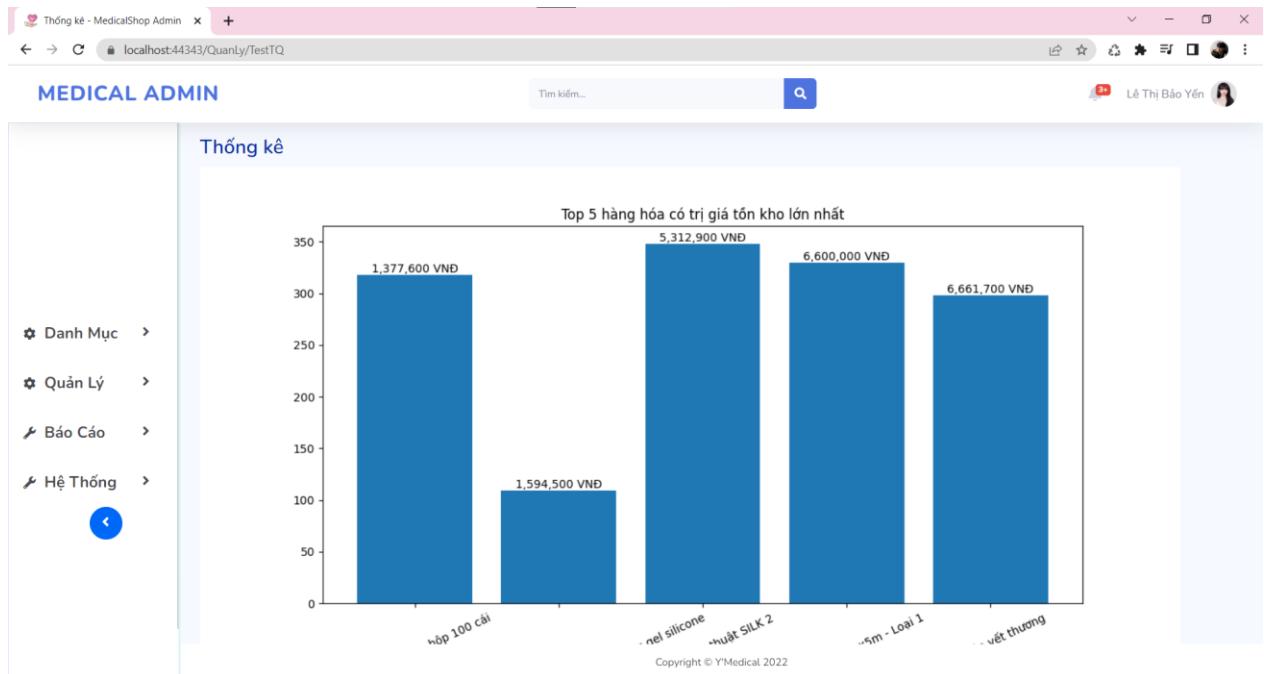
    plt.rcParams["figure.figsize"] = (12, 6) # đổi kích thước biểu đồ
    fig, ax = plt.subplots()
    bars = ax.bar(data["tenHH"], data["sl"]) # đổi thành bar chart
    labels = [f'{value:,.0f} VNĐ' for value in data['gia']]
    ax.bar_label(bars, labels=labels) # hiển thị giá trị của các cột lên trên chúng
    plt.title("Top 5 hàng hóa có trị giá tồn kho lớn nhất")
    plt.xticks(rotation=25)
    display(fig, target="graph-area", append=False)

data = pd.DataFrame(data)
plot(data)
```

</py-script>

- Khởi tạo hàm plot
 - Thiết lập kích thước cho biểu đồ.
 - Khởi tạo fig và ax bằng phương thức subplots().
 - Khởi tạo biểu đồ hình cột với trục x là 'tenHH' và trục y là 'sl'.
 - Hiển thị giá trị trên các cột ép sang kiểu tiền VNĐ.
 - Điều chỉnh tiêu đề của biểu đồ.
 - Nhãn tên đơn vị nghiêng 45⁰ để giảm độ dính tên.
 - Hiển thị fig, với append=False thì fig sẽ được hiển thị đè lên vùng hiển thị đã có sẵn ở đây là 'graph-area'.

- Kết quả














Hình 3. 3 Trực quan với Pyscript

3.2.4 Ứng dụng

Xây dựng một công cụ cơ bản, khi người dùng có một table được lấy ở dạng HTML, người dùng có thể chuyển dữ liệu của bảng về dạng Json hoặc Excel để thu thập dữ liệu và sử dụng.

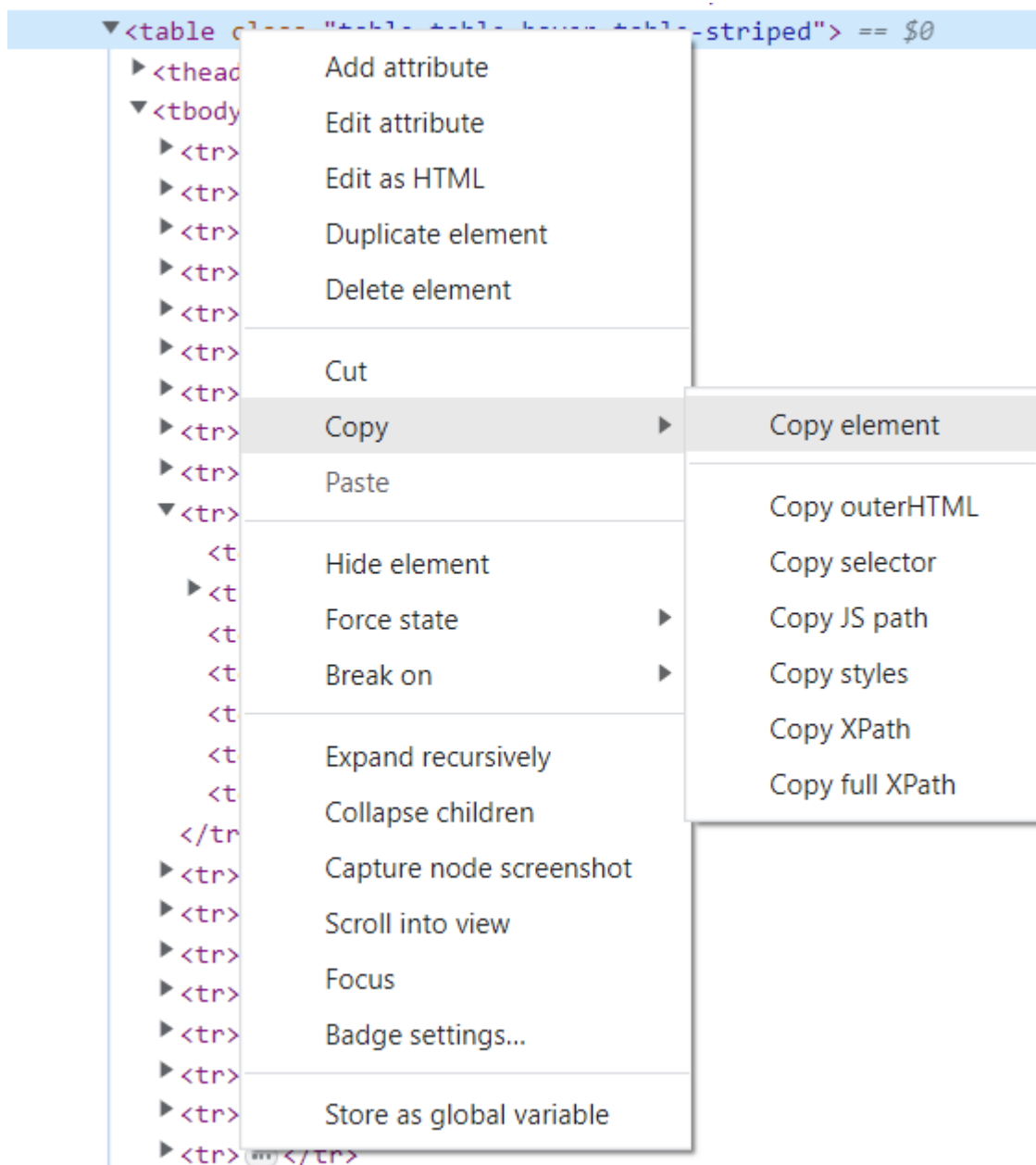
Với trang web muốn thu thập như sau:

STT	TÊN VẬT TƯ Y TẾ	MÃ SẢN PHẨM	PHÂN LOẠI VTYT THEO MỨC ĐỘ RỦI RO	NHÓM VTYT(TT14/2020/TT-BYT)	HÃNG CHỦ SỞ HỮU	GIÁ NIÊM YẾT (VNĐ)
1	 Vật tư tiêu hao dùng cho máy phân tích miễn dịch	CP815570	1	Nhóm 6	Sysmex Corporation	8.102.001
2	 Ống phản ứng sử dụng trên máy đông máu tự động	90407219	1	Nhóm 3	Sysmex Corporation	14.568.400
3	 Cổng đo thực hiện xét nghiệm ngưng tập tiểu cầu trên hệ thống đông máu tự động	06410419	1	Nhóm 6	Sysmex Corporation	11.465.377
4	 Vật tư tiêu hao dùng cho máy phân tích đông máu	06414810	1	Nhóm 6	Sysmex Corporation	17.490.000
5	 Cốc đựng mẫu, hóa chất	42411608	1	Nhóm 6	Sysmex Corporation	1.749.003
6	 Giếng phản ứng sử dụng trên máy đông máu bán tự động	AG405069	1	Nhóm 6	LABiTec LAbor BioMedical Technologies GmbH	5.720.000
7	 Đỉnh schanz đường kính các loại	EF-209xx xxxxxx	3	Nhóm 3	MAT GmbH & Co.KG	300.000
8	 Đỉnh kit ne đường kính các loại	004-0310-xxx	3	Nhóm 3	MAT GmbH & Co.KG	100.000
9	 Chỉ thép mềm đường kính các loại (cuộn 10m)	INS-8091-xx	3	Nhóm 3	MAT GmbH & Co.KG	800.000
10	 Nẹp mắt xích các cỡ	PL-1025-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.000.000
11	 Nẹp chữ L trái, phải, vít 4.5mm	PL-1038-xxx	3	Nhóm 3	MAT GmbH & Co.KG	1.600.000

Dữ liệu đang được trình bày ở dạng bảng, cấu trúc HTML như sau:

[illegible]

Tại đây, người dùng có thể sao chép dữ liệu HTML một cách nhanh chóng bằng cách sau:



Lúc này, người dùng chỉ cần truy cập vào trang web và dán vào hộp văn bản. Sau đó nhấn chuyển đổi để thực hiện quá trình chuyển đổi dữ liệu từ HTML.

Sau khi quá trình chuyển đổi hoàn tất, màn hình sẽ hiển thị giá trị Json sau khi đã chuyển đổi và hiển thị sang bảng dữ liệu bên dưới. Tại đây, người dùng có thể lưu dữ liệu về dạng Json hoặc Excel.

MedicalShop Admin

localhost:44343/QuanLy/ConvertTableToData

🔍 🏠 ⚙️ 📄 📱 🖨️

MEDICAL ADMIN

Tìm kiếm...

🔍

Lê Thị Bảo Yến

🔍

Danh Mục

🔍

Quản Lý

🔍

Báo Cáo

🔍

Hệ Thống

Chuyển đổi

Excel

Json

Table HTML

Json Data

page=Project.MedicalPrice.Home.MedicalPrice.Material.detail&id=611f78afffd9a12984a68f2&categoryId=5f9bc540c5dbc2404e1d7e23" title="Chi tiết">Xem chi tiết<i class="fas fa-arrow-right pl-10"></i> </div> </div> </div> <div class="tb-title"> Nep bản nhỏ các cơ </div> </div> </td> <td>PL-1017-

STT	Tên vật tư y tế	Mã sản phẩm	Phân loại VTYT theo mức độ rủi ro	Nhóm VTYT(TT14/2020/TT-BYT)	Hãng chủ sở hữu	Giá niêm yết (VNĐ)
1	Xem chi tiết Vật tư tiêu hao dùng cho máy phân tích miễn dịch	CP815570	1	Nhóm 6	Sysmex Corporation	8.102.001
2	Xem chi tiết Ống phản ứng sử dụng trên máy đồng mẫu tự	90407219	1	Nhóm 3	Sysmex Corporation	14.568.400

Copyright © YMedical 2022

Hình 3. 4 Kết quả thu thập được của công cụ tư tạo

Kết quả tải về:

```
[
  {
    "STT": "1",
    "Tên vật tư y tế": "Xem chi tiết      Vật tư tiêu hao dùng cho máy phân tích
miễn dịch",
    "Mã sản phẩm": "CP815570",
    "Phân loại VTYT theo mức độ rủi ro ": "1",
    "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
    "Hãng chủ sở hữu ": "Sysmex Corporation",
    "Giá niêm yết (VNĐ)": "8.102.001"
  },
  {
    "STT": "2",
    "Tên vật tư y tế": "Xem chi tiết      Ống phản ứng sử dụng trên máy đông máu
tự động",
    "Mã sản phẩm": "90407219",
    "Phân loại VTYT theo mức độ rủi ro ": "1",
    "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 3",
    "Hãng chủ sở hữu ": "Sysmex Corporation",
    "Giá niêm yết (VNĐ)": "14.568.400"
  },
  {
    "STT": "3",
    "Tên vật tư y tế": "Xem chi tiết      Công đo thực hiện xét nghiệm ngưng tập
tiểu cầu trên hệ thống đông máu tự động",
    "Mã sản phẩm": "06410419",
    "Phân loại VTYT theo mức độ rủi ro ": "1",
    "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
    "Hãng chủ sở hữu ": "Sysmex Corporation",
    "Giá niêm yết (VNĐ)": "11.465.377"
  },
  {
    "STT": "4",
    "Tên vật tư y tế": "Xem chi tiết      Vật tư tiêu hao dùng cho máy phân tích
đông máu",
    "Mã sản phẩm": "06414810",
    "Phân loại VTYT theo mức độ rủi ro ": "1",
    "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
    "Hãng chủ sở hữu ": "Sysmex Corporation",
    "Giá niêm yết (VNĐ)": "17.490.000"
  },
  {
    "STT": "5",
    "Tên vật tư y tế": "Xem chi tiết      Cốc đựng mẫu, hóa chất",
    "Mã sản phẩm": "42411608",
    "Phân loại VTYT theo mức độ rủi ro ": "1",
```











Hiện tại vẫn có lỗi ở key: “Tên vật tư y tế”. Tiến hành làm sạch dữ liệu, kết quả:

```
{
  "STT": "1",
  "Tên vật tư y tế": "Vật tư tiêu hao dùng cho máy phân tích miễn dịch",
  "Mã sản phẩm": "CP815570",
  "Phân loại VTYT theo mức độ rủi ro ": "1",
  "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
  "Hãng chủ sở hữu ": "Sysmex Corporation",
  "Giá niêm yết (VNĐ)": "8.102.001"
},
{
  "STT": "2",
  "Tên vật tư y tế": "Ống phản ứng sử dụng trên máy đông máu tự động",
  "Mã sản phẩm": "90407219",
  "Phân loại VTYT theo mức độ rủi ro ": "1",
  "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 3",
  "Hãng chủ sở hữu ": "Sysmex Corporation",
  "Giá niêm yết (VNĐ)": "14.568.400"
},
{
  "STT": "3",
  "Tên vật tư y tế": "Cống đo thực hiện xét nghiệm ngưng tập tiểu cầu trên hệ thống đông máu tự động",
  "Mã sản phẩm": "06410419",
  "Phân loại VTYT theo mức độ rủi ro ": "1",
  "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
  "Hãng chủ sở hữu ": "Sysmex Corporation",
  "Giá niêm yết (VNĐ)": "11.465.377"
},
{
  "STT": "4",
  "Tên vật tư y tế": "Vật tư tiêu hao dùng cho máy phân tích đông máu",
  "Mã sản phẩm": "06414810",
  "Phân loại VTYT theo mức độ rủi ro ": "1",
  "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
  "Hãng chủ sở hữu ": "Sysmex Corporation",
  "Giá niêm yết (VNĐ)": "17.490.000"
},
{
  "STT": "5",
  "Tên vật tư y tế": "Cốc đựng mẫu, hóa chất",
  "Mã sản phẩm": "42411608",
  "Phân loại VTYT theo mức độ rủi ro ": "1",
  "Nhóm VTYT(TT14/2020/TT-BYT) ": "Nhóm 6",
  "Hãng chủ sở hữu ": "Sysmex Corporation",
  "Giá niêm yết (VNĐ)": "1.749.003"
},
{
```

Dữ liệu được lưu về ở dạng Excel

STT	Tên vật tư y tế	Mã sản phẩm	Phân loại VTYT	Nhóm VTYT	Hãng chủ sở hữu	Giá niêm yết (VNĐ)
1	Vật tư tiêu hao dùng cho máy phân tích miễn dịch	CP815570	1	Nhóm 6	Sysmex Corporation	8.102.001
2	Ống phản ứng sử dụng trên máy đồng mẫu	190407219	1	Nhóm 3	Sysmex Corporation	14.568.400
3	Công đo thực hiện xét nghiệm ngưng tập tiểu cầu	06410419	1	Nhóm 6	Sysmex Corporation	11.465.377
4	Vật tư tiêu hao dùng cho máy phân tích đông máu	06414810	1	Nhóm 6	Sysmex Corporation	17.490.000
5	Cốc đựng mẫu, hóa chất	42411608	1	Nhóm 6	Sysmex Corporation	1.749.003
6	Giếng phản ứng sử dụng trên máy đồng mẫu	AG405069	1	Nhóm 6	LABiTec Labor BioMedical Tech	5.720.000
7	Đinh schanz đường kính các loại	EF-209xx xxxxxx	3	Nhóm 3	MAT GmbH & Co.KG	300.000
8	Đinh kit ne đường kính các loại	004-0310-xxx	3	Nhóm 3	MAT GmbH & Co.KG	100.000
9	Chỉ thép mềm đường kính các loại (cuộn 10m)	INS-8091-xx	3	Nhóm 3	MAT GmbH & Co.KG	800.000
10	Nẹp mắt xích các cỡ	PL-1025-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.000.000
11	Nẹp chữ L trái, phải, vít 4.5mm	PL-1038-xxx	3	Nhóm 3	MAT GmbH & Co.KG	1.600.000
12	Nẹp chữ T, vít 4.5mm	PL-1035-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.600.000
13	Nẹp chữ T nhỏ 3 lỗ đầu (3 - 6 lỗ thân) vít 3.5mm	PL-1043-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.000.000
14	Nẹp bản rộng các cỡ	PL-1011-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.300.000
15	Nẹp bản hẹp các cỡ	PL-1013-xx	3	Nhóm 3	MAT GmbH & Co.KG	1.200.000
16	Nẹp bản nhỏ các cỡ	PL-1017-xx	3	Nhóm 3	MAT GmbH & Co.KG	900.000
17	Nẹp lòng máng 1/3, vít 3.5 mm	PL-1020-xx	3	Nhóm 3	MAT GmbH & Co.KG	550.000
18	Vít mắt cá chân đường kính 4.5mm	001-0009-xxx	3	Nhóm 3	MAT GmbH & Co.KG	250.000
19	Vi ống thông Asahi Veloute	VEL105-16S, VEL105-16.4	3	Nhóm 4	Asahi Intecc Co., Ltd.	12.500.000
20	Vi ống thông Asahi Tellus	TLS105-16S, TLS105-16.4	3	Nhóm 4	Asahi Intecc Co., Ltd.	12.500.000

Trương tự, thử với dữ liệu của trang web quản lý vật tư y tế:

Mã NHH	Tên NHH	Ngày tạo	NV tạo	Ngày sửa	NV sửa	Tùy chọn
NHOM1	Bông, dung dịch sát khuẩn, rửa vết thương	15-12-2022 23:12	Lê Thị Bảo Yến	15-12-2022 23:12	Lê Thị Bảo Yến	 
NHOM2	Băng, gạc, vật liệu cầm máu, điều trị vết thương	15-12-2022 23:12	Lê Thị Bảo Yến	15-12-2022 23:12	Lê Thị Bảo Yến	 
NHOM3	Bơm, kim tiêm, dây truyền, găng tay và vật tư y tế sử dụng trong chăm sóc người bệnh	15-12-2022 23:24	Lê Thị Bảo Yến	31-01-2023 15:15	Lê Thị Bảo Yến	 
NHOM4	Ống thông, ống dẫn lưu, ống nối, dây nối, chạc nối, catheter	31-01-2023 15:18	Lê Thị Bảo Yến	31-01-2023 15:22	Lê Thị Bảo Yến	 
NHOM5	Kim khâu, chỉ khâu, dao phẫu thuật	11-05-2023 20:56	Lê Thị Bảo Yến	11-05-2023 20:56	Lê Thị Bảo Yến	 

Kết quả thu được như sau:

The screenshot shows a web application interface for 'MEDICAL ADMIN'. It features a sidebar with navigation links: 'Danh Mục', 'Quản Lý', 'Báo Cáo', and 'Hệ Thống'. The main content area displays a table of medical data with columns: 'Mã NHH', 'Tên NHH', 'Ngày tạo', 'NV tạo', 'Ngày sửa', 'NV sửa', and 'Tùy chọn'. The table contains three rows of data. To the right of the table, there is a 'Json Data' section showing the corresponding JSON representation of the table data. Below the table, there are buttons for 'Chuyển đổi', 'Excel', and 'Json'.

Mã NHH	Tên NHH	Ngày tạo	NV tạo	Ngày sửa	NV sửa	Tùy chọn
NHOM1	Bông, dung dịch sát khuẩn, rửa vết thương	15-12-2022 23:12	Lê Thị Bảo Yến	15-12-2022 23:12	Lê Thị Bảo Yến	
NHOM2	Băng, gạc, vật liệu cầm máu, điều trị vết thương	15-12-2022 23:12	Lê Thị Bảo Yến	15-12-2022 23:12	Lê Thị Bảo Yến	
NHOM3	Bơm, kim tiêm, dây truyền, găng tay và vật tư y tế sử dụng trong chăm sóc người bệnh	15-12-2022 23:24	Lê Thị Bảo Yến	31-01-2023 15:15	Lê Thị Bảo Yến	

3.3 Một số nghiên cứu khác

3.3.1 Tạo tự động dữ liệu mẫu

Xây dựng một hệ thống website quản lý vật tư y tế, cần số lượng dữ liệu lớn để kiểm tra chất lượng hệ thống, việc tạo tay gây tốn thời gian. Nghiên cứu tạo dữ liệu tự động bằng Python.

- Thư viện
 - Thư viện Faker để tạo dữ liệu giả.
 - Thư viện Random để lấy giá trị ngẫu nhiên.
 - Module Workbook của thư viện openpyxl để tạo và xử lý file Excel.
 - Thư viện os dùng để thao tác với thư mục lấy địa chỉ lưu file.
 - Module date của thư viện Datetime để xử lý thời gian của ngày sinh.
 - Callable, Any để định nghĩa kiểu dữ liệu đầu vào là Faker và trả về là List bất kỳ.
 - BaseProvider để tùy chỉnh các giá trị giả mạo.
- Thực nghiệm

Tạo dữ liệu giả cho bảng Nhân Viên gồm các thuộc tính sau: ['ID', 'Ten', 'CCCD', 'GioiTinh', 'NgaySinh', 'QueQuan', 'DiaChi', 'SDT', 'Mail', 'ChiNhanh'].

- Tạo Random cho ID

Tạo ngẫu nhiên chuỗi với độ dài là 6 kí tự số.

```
class RandomChar(BaseProvider):
    def generate_random_id(self) -> str:
        return ''.join(str(random.randint(1, 999999)).zfill(6))
```

b. Tùy chỉnh giá trị ChiNhanh

Giả sử hệ thống quản lý có 3 ChiNhanh là CN01, CN02 và CN03, tùy chỉnh sẽ lấy một trong 3 lựa chọn.

```
class ChiNhanh(BaseProvider):
    def generate_CN(self) -> str:
        branches = ['CN01', 'CN02', 'CN03']
        return random.choice(branches)
```

c. Định dạng kiểu điện thoại vùng VN

Với số điện thoại di động là +84##### và +84##### tương ứng với số điện thoại bàn ở Việt Nam. Tạo số điện thoại giả sẽ random số ở kí tự #.

```
class VNPhoneNumber(BaseProvider):
    def vn_phone_number(self):
        formats = ['+84#####', '+84#####'] #sdt di dong, sdt
        return self.numerify(self.random_element(formats))
```

d. Xây dựng hàm tạo dữ liệu nhân viên sử dụng Faker trả về List kiểu string.

Trong đó có kiểm tra ngày sinh của nhân viên, nếu ngày sinh trước 1/1/1972 hoặc sau 1/1/2002 thì tạo lại ngày sinh, có thể tùy chỉnh lại. Sau đó là lệnh Return trả về với các dòng lần lượt là:

- Tạo ID sử dụng hàm generate_random_id()
- Tên nhân viên
- CCCD với 12 số
- Random giới tính kiểu 0-1 để phù hợp với dữ liệu kiểu bit trong database, với 0 là nam và 1 là nữ.

- Lấy ngày sinh đã tạo tự động và phù hợp điều kiện ở trên
- Tạo Quê quán
- Tạo địa chỉ
- Tạo số điện thoại sử dụng hàm vn_phone_number()
- Tạo email
- Tạo Chi Nhánh làm việc với hàm generate_CN()

```
def person_data_generator(faker: Faker) -> list[str]:
    # Kiểm tra nếu ngày sinh trước 1/1/1972 hoặc sau 1/1/2002 thì tạo lại ngày sinh
    dob = faker.date_of_birth()
    while dob <= date(1972, 1, 1) or dob >= date(2002, 1, 1):
        dob = faker.date_of_birth()

    return [faker.generate_random_id(), #ma faker.generate_random_id()
            faker.name(), #ten
            str(random.randint(100000000, 999999999)).zfill(12), #cccd
            random.randint(0, 1), #gioitinh
            dob, #ngaysinh
            faker.country(),
            faker.address(),
            faker.vn_phone_number(),
            faker.email(),
            faker.generate_CN()]
```

e. Xây dựng hàm tạo file Excel

Hàm bao gồm khai báo filename, tổng số record muốn tạo tự động, hàm với đối số là Faker trả về kiểu dữ liệu List, là header của file. Khởi tạo đường dẫn lưu file và tạo workbook với các dòng dữ liệu thêm sau vào cho tới hết số record.

```
def generate_excel(filename: str,
                  row_num: int,
                  generator: Callable[[Faker], list[Any]],
                  faker: Faker,
                  header: list[str]) -> None:
    os.makedirs(os.path.dirname(filename), exist_ok=True)
    wb = Workbook()
    ws = wb.active
    ws.append(header)
    for n in range(1, row_num):
        ws.append(generator(faker))
    wb.save(filename)
```

f. Xây dựng hàm sinh dữ liệu giả

Khởi tạo Faker, Add Provider các Class chứa hàm generate đã tùy chỉnh. Khởi tạo workbook và worksheet, thiết lập header và số record. Sau đó tạo dữ liệu bằng hàm `person_data_generator()` đã tạo trước đó, cuối cùng lưu file excel với tên là `danh sach nhan vien` tại thư mục gốc.

```
def fake_data_plan() -> None:
    faker: Faker = Faker()
    faker.add_provider(RandomChar)
    faker.add_provider(ChiNhanh)
    faker.add_provider(VNPhoneNumber)

    # Tạo workbook mới
    wb = Workbook()
    # Tạo worksheet mới
    ws = wb.active

    # Thiết lập header rows
    header_rows = ['ID', 'Ten', 'CCCD', 'GioiTinh', 'NgaySinh', 'QueQuan', 'DiaChi', 'SDT', 'Mail', 'ChiNhanh']
    ws.append(header_rows)

    # Thiết lập số dòng cần tạo
    row_num = 1000

    # Tạo dữ liệu và ghi vào worksheet
    for n in range(row_num):
        row_data = person_data_generator(faker)
        ws.append(row_data)

    # Lưu workbook vào file
    filename = os.path.join(os.getcwd(), 'danh sach nhan vien.xlsx')
    wb.save(filename)
```


- Kết quả

	A	B	C	D	E	F	G	H	I	J	K	L
1	ID	Ten	CCCD	GioiTinh	NgaySinh	QueQuan	DiaChi	SDT	Mail	ChiNhanh		
2	720679	Lisa Vaughn	000611579174	1	1978-01-26	Nicaragua	061 James ValleysCarohview, VT 21837	+84396958732	baileybrian@example.net	CN03		
3	084972	Krista Perry	000153942402	1	1986-10-26	New Zealand	1918 Beck Village Suite 757Michaelmouth, DC 38321	+84649963605	ocurtis@example.net	CN02		
4	743560	Megan Hill	000709254790	1	1993-05-13	Angola	3510 Paul RoadsNorth Christopher, DE 33604	+841565838	jacob37@example.org	CN03		
5	440487	Susan Norris	00073645787	1	1972-10-13	Seychelles	506 Zamora Flats Suite 063South William, ME 94674	+84444825091	lyonslaura@example.net	CN03		
6	779668	Robert Washington	000831625740	1	1980-07-03	Niue	6629 Jennifer Mill Apt. 094Port Shawnborough, PW 33564	+84984261446	paulshah@example.org	CN03		
7	653131	Anne Smith	000262303005	1	2001-07-21	United States Virgin Islands	Unit 8795 Box 0847DPO AE 11317	+84867119304	yross@example.org	CN03		
8	527232	Marc Lee	000304821387	0	1973-11-07	Paraguay	117 Tonya GardenNorth Kevinberg, AK 99062	+849648477	mballard@example.com	CN03		
9	829145	Joseph Schmidt	000408368800	0	1983-05-14	Togo	6478 Samuel Extension Suite 966North Karen, WA 78373	+844729645	justinmassey@example.org	CN01		
10	271853	Danielle Leonard	000272338864	0	1979-02-02	French Guiana	44955 Miller Court Suite 131South Seanberg, AS 14231	+846891240	carrollmckenzie@example.org	CN02		
11	075431	Debra Williams	000646315817	0	1972-10-19	Kuwait	34901 Gomez Curve Suite 469Danashire, SC 49756	+849421414	kevin21@example.net	CN01		
12	006797	Christopher Scott	000339943256	0	1992-03-03	Guinea	4745 Erin PrairieChristopherchester, MA 27268	+844271336	jacqueline14@example.org	CN01		
13	300689	Vincent Gonzalez	000620893564	1	1996-07-17	Netherlands	536 Hopkins Parkway Suite 570Williamsport, WY 19658	+84801501084	christophermore@example.com	CN01		
14	712243	Judy Thomas	000369397588	1	1994-12-20	Comoros	630 Wright Centers Suite 352East Erin, MI 94423	+847315887	sking@example.com	CN01		
15	451601	Nathan Cox	000642450632	1	2001-07-24	Slovenia	12210 Daniel Port Suite 020Kurtville, TX 89263	+846775991	lukelloyd@example.org	CN01		
16	906651	Lori Pierce	000710049848	0	1977-03-29	Tokelau	149 Humphrey Forks Apt. 3795Smithshire, FM 72178	+849016009	zlopez@example.org	CN02		
17	909200	Amanda Stewart	000799048974	1	1992-02-01	Venezuela	0175 Proctor Green Apt. 464Sandriaview, SC 37549	+847656705	stephaniehebert@example.org	CN01		
18	527677	Margaret Christian	000438298910	0	1976-07-06	Monaco	1195 Jasmine SummitNew Kevin, AK 24898	+841692415	vanessa52@example.org	CN01		
19	176630	Jeremiah Johnson	000634410222	0	1986-04-10	Cocos (Keeling) Islands	302 Simmons SquaresNorth Amanda, ND 66526	+84778479736	danielbrown@example.net	CN03		
20	079047	Melissa Snow	000267761937	0	1990-04-03	Cameroon	474 Jack Bridge Apt. 015Johnsonborough, MP 42432	+841113969	joseph42@example.com	CN03		
21	904896	Angela Branch	000195004294	0	1987-06-10	Norway	3321 Christy StationPort David, VA 26218	+842147620	foleysabrina@example.org	CN01		
22	540559	Renee Williams	000426503658	1	1996-09-22	Ecuador	252 Harris RanchEast Lisa, MH 27901	+84656193255	gallagherwilliam@example.net	CN02		
23	202156	Matthew Smith	000655835864	1	1973-11-27	Saint Helena	USNS MoodyFPO AA 46475	+843159437	gamblealexander@example.net	CN01		
24	096600	Steve Barnett	000756532232	1	1999-08-08	Saint Pierre and Miquelon	39758 Tonya SkywayHollowayside, IL 85841	+84177566771	mcherry@example.net	CN02		
25	207129	Desiree Miller	000742311926	1	1982-02-02	French Southern Territories	5584 Haley HillPalmermouth, ME 69892	+84718871682	ibarrarebecca@example.net	CN03		
26	490000	David Coleman	000339772121	0	1991-03-04	Bermuda	868 Harris ClubMcdonaldfort, NH 09554	+842216306	dayronald@example.net	CN02		
27	426754	Nathan Guzman	000147343482	1	1990-04-17	Bangladesh	4802 Castro Brooks Suite 375Anthonyemouth, PR 11364	+84032456004	tjensen@example.net	CN02		
28	270714	Patricia Patton	000967227969	1	1974-09-28	Kyrgyz Republic	6533 Farmer ShoalNorth Victoria, NY 61133	+846463281	douglascharles@example.com	CN03		
29	078916	Nicholas Payne	000230859673	1	1995-11-17	Marshall Islands	94666 Walker FlatEvansview, MD 80074	+845673557	saraevans@example.org	CN02		
30	386506	Timothy Ray	000124284653	0	1983-03-29	United Arab Emirates	3632 Mackenzie StravenueLeepport, ND 76674	+84317965793	briggsdoris@example.org	CN02		

3.3.2 Thêm dữ liệu tự động

Với data.json tự lấy được tại 3.2.4 có thể tiến hành thêm vào cơ sở dữ liệu để sử dụng, tuy nhiên với số lượng khá lớn, việc nhập tay không khả quan. Từ đó em tiến hành nghiên cứu việc lưu trữ dữ liệu lớn một cách tự động để sử dụng.

- Cài đặt gói Newtonsoft.Json.
- Sử dụng Entity Framework nên cài thêm
 - Microsoft.EntityFrameworkCore.SqlServer 3.1.30
 - Microsoft.EntityFrameworkCore.Design 3.1.30
 - Microsoft.EntityFrameworkCore.Tools 3.1.30

- Khởi tạo một hàm tudong().

```
private void tudong(){
    string jsonFilePath = @"C:\Users\Yuta\Downloads\ngheNghiep.json";
    string jsonContent = System.IO.File.ReadAllText(jsonFilePath);
    JArray jsonArray = JArray.Parse(jsonContent);
    foreach (JObject jsonObject in jsonArray)
    {
        string tenNgheNghiep = jsonObject["Tên gọi nghề nghiệp"].ToString();
        using (var context = new HTRQDContext())
        {
            var ngheNghiep = new NgheNghiep {TenNn = tenNgheNghiep };
            context.NgheNghiep.Add(ngheNghiep);
            context.SaveChanges();
        }
    }
}
```

Trong đó, gọi ra đường dẫn của file Json. Đọc toàn bộ nội dung từ file JSON. Phân tích cú pháp JSON. Chạy vòng lặp, với mỗi jsonObject sẽ lấy value với key là: “Tên gọi nghề nghiệp” (json là các cặp key: value). Sử dụng Entity Framework để thêm dữ liệu.

- Kết quả

34
35 | SELECT n.TenNN FROM NgheNghiep n

133 %

Results Messages

	TenNN
1	Bác sỹ chuyên khoa khác
2	Bác sỹ đa liệu
3	Bác sỹ nội khoa
4	Bác sỹ ngoại khoa
5	Bác sỹ nhi khoa
6	Bác sỹ răng - hàm - mặt
7	Bác sỹ thú y
8	Bán buôn, bán lẻ
9	Bán hàng và tiếp thị
10	Biên dịch
11	Biên tập viên
12	Biên tập viên xuất bản phẩm
13	Bồi bản
14	Bồi bản (trừ bồi bản rượu)
15	Bồi bản rượu
16	Bồi bản và nhân viên pha chế
17	Ca sỹ

Query executed successfully. | LAPTOP-J4VFA095 (15.0 RTM) | LAPTOP-J4VFA095\Yuta (59) | HTRQD 00:00:00 | 713 rows

3.4 Kết quả đạt được

Tìm hiểu được các phương pháp thu thập dữ liệu, làm sạch dữ liệu. Tìm hiểu được nhiều phương pháp trực quan hóa dữ liệu, từ đó đưa ra lựa chọn phù hợp tùy với nhu cầu sử dụng.

Tạo ra công cụ nhỏ để hỗ trợ quá trình thu thập dữ liệu nhanh hơn, lưu trữ sử dụng và quản lý dữ liệu.

CHƯƠNG IV: KẾT LUẬN

4.1. Kết luận

4.1.1. Đánh giá kết quả đạt được

4.1.1.1. Ưu điểm

Có thể thu nhập nguồn dữ liệu lớn từ trang web của người khác bằng cách chuyển đổi dữ liệu ở dạng HTML sang Json để sử dụng. Sẽ làm sạch nguồn dữ liệu sau khi thu thập. Sau khi dữ liệu được làm sạch, đã nghiên cứu được cách thêm tự động nguồn dữ liệu trên vào cơ sở dữ liệu để tiến hành sử dụng và lưu trữ.

Ngoài ra, trong quá trình nghiên cứu còn tìm hiểu cách thức tạo tự động dữ liệu, từ đó có thể ứng dụng để tạo nguồn dữ liệu giả lớn để sử dụng vào mục đích kiểm tra hệ thống.

4.1.1.2. Nhược điểm

def.

4.1.2. Kiến thức đạt được

Kỹ thuật thu thập dữ liệu từ trang web sử dụng các công cụ và thư viện như BeautifulSoup và Selenium.

Các bước làm sạch dữ liệu bao gồm chuyển đổi kiểu dữ liệu, loại bỏ ký tự không cần thiết và xử lý giá trị trống hoặc không hợp lệ.

Chuyển đổi dữ liệu từ định dạng bảng HTML sang định dạng JSON để thuận tiện trong việc lưu trữ, truyền tải và sử dụng dữ liệu.

Các phương pháp và công cụ phân tích dữ liệu cơ bản để hiểu và khám phá thông tin từ dữ liệu thu thập được.

Tổng kết, qua đồ án này, chúng ta đã áp dụng thành công quy trình thu thập, phân tích và chuyển đổi dữ liệu từ trang web đưa table ở dạng HTML về kiểu JSON. Đồ án này đã cung cấp cho chúng ta kiến thức và kỹ năng để nghiên cứu, làm sạch và ứng dụng dữ liệu từ các nguồn khác nhau trong thực tế.

4.2. Hướng phát triển

Mở rộng khả năng thu thập dữ liệu: Nghiên cứu và thực hiện thu thập dữ liệu từ nhiều trang web khác nhau đưa table ở dạng HTML và chuyển đổi chúng thành định dạng

JSON. Điều này mở ra khả năng thu thập dữ liệu từ nhiều nguồn khác nhau và mở rộng phạm vi ứng dụng của đề tài.

Tăng cường quy trình làm sạch dữ liệu: Nghiên cứu và áp dụng các phương pháp và công cụ để làm sạch dữ liệu thu thập được từ trang web. Bạn có thể áp dụng các thuật toán xử lý ngôn ngữ tự nhiên để tách từ, loại bỏ stop words và phân loại các thuộc tính dữ liệu.

Phân tích và khai phá dữ liệu: Áp dụng các kỹ thuật và công cụ phân tích dữ liệu để tìm hiểu sâu hơn về thông tin vật tư y tế. Bạn có thể thực hiện phân tích đa biến, phân tích độ tuổi khách hàng, phân tích xu hướng và dự đoán giá trị, hoặc phát hiện nhóm khách hàng tiềm năng.

Tích hợp với công cụ hỗ trợ quyết định: Phát triển một công cụ hỗ trợ quyết định dựa trên dữ liệu thu thập được. Ví dụ, có thể tạo mô hình dự đoán giá trị vật tư dựa trên các thuộc tính và xây dựng một công cụ cho phép người dùng dự đoán giá trị dựa trên thông tin đã cho.

TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1] **Atal Malviya, Mike Malmgren**, *Big data cho nhà quản lý: Cuốn sách gối đầu giường cho mọi CEO - THB*, Công thương, 2020
- [2] **Bernard Marr**, *Data Strategy - Chiến Lược Dữ Liệu fs*, NXB Tổng Hợp TP.HCM, 2019
- [3] **Nhiều tác giả**, *Dữ Liệu Lớn – Cuộc Cách Mạng Thay Đổi Chúng Ta Và Thế Giới*, Khoa Học Kỹ Thuật, 2020.

Tiếng Anh:

- [1] **Adam Freeman và Steve Sanderson**, *Pro ASP.NET MVC 4*, Apress December 21, 2012. **V. Chandola, A. Banerjee, và V. Kumar**, "Anomaly Detection: A Survey," trong *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 11, pp. 1484-1504, 2009.
- [2] **J. Han, M. Kamber, và J. Pei**, "**Data Mining: Concepts and Techniques**," 3rd ed., Elsevier, 2011.
- [3] **L. Breiman**, "**Random Forests**," trong *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.