A blue and red text on a black background

AI-generated content may be incorrect.BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

KHOA CÔNG NGHỆ THÔNG TIN

**ĐỒ ÁN MÔN HỌC**

**PHÂN TÍCH VÀ ĐỀ XUẤT QUÁN ĂN DỰA TRÊN DỮ LIỆU ĐÁNH GIÁ NGƯỜI DÙNG TỪ FOODY.VN SỬ DỤNG CÔNG CỤ SELENIUM VÀ MONGODB**

Ngành: **KHOA HỌC DỮ LIỆU**

Môn học: **MÃ NGUỒN MỞ**

Giảng viên hướng dẫn : ThS. Lê Nhật Tùng

Sinh viên thực hiện : 2386400768-Nguyễn Thị Sương Anh

2386400003-Nguyễn Thị Kiều Anh

Lớp: 23DKHA1

TP. Hồ Chí Minh, 2025

A blue and red text on a black background

AI-generated content may be incorrect.BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

KHOA CÔNG NGHỆ THÔNG TIN

**ĐỒ ÁN MÔN HỌC**

**PHÂN TÍCH VÀ ĐỀ XUẤT QUÁN ĂN DỰA TRÊN DỮ LIỆU ĐÁNH GIÁ NGƯỜI DÙNG TỪ FOODY.VN SỬ DỤNG CÔNG CỤ SELENIUM VÀ MONGODB**

Ngành: **KHOA HỌC DỮ LIỆU**

Môn học: **MÃ NGUỒN MỞ**

Giảng viên hướng dẫn : ThS. Lê Nhật Tùng

Sinh viên thực hiện : 2386400768-Nguyễn Thị Sương Anh

2386400003-Nguyễn Thị Kiều Anh

Lớp: 23DKHA1

TP. Hồ Chí Minh, 2025

# **NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... ....................................................................................................................... .......................................................................................................................

....................................................................................................................... .......................................................................................................................

....................................................................................................................... .......................................................................................................................

....................................................................................................................... .......................................................................................................................

TPHCM, Ngày…... Tháng …... Năm 2025

Giáo viên hướng dẫn

(Ký tên, đóng dấu)

# **LỜI CAM ĐOAN**

Nhóm chúng tôi gồm Nguyễn Thị Sương Anh, Nguyễn Thị Kiều Anh xin cam đoan rằng:

Tất cả thông tin và kết quả nghiên cứu trong bài báo cáo này đều trung thực và khách quan, được thu thập từ các nguồn đáng tin cậy, chính thống. Chúng tôi đã phân tích kỹ lưỡng các tài liệu và đảm bảo rằng mọi dữ liệu hoặc ý kiến trích dẫn đều được ghi rõ ràng nguồn gốc, tuân thủ đúng quy định về trích dẫn học thuật.

Chúng tôi cam kết không có hành vi sao chép hoặc sử dụng trái phép bất kỳ thông tin nào từ các nguồn khác. Bài báo cáo này là sản phẩm nghiên cứu độc lập của nhóm, chưa từng được công bố trước đây và tuân thủ đầy đủ các quy định của môn học. Chúng tôi sử dụng công cụ nghiên cứu một cách hợp lý và chính xác, đảm bảo tính khoa học và đạo đức trong quá trình thực hiện.

TPHCM, Ngày…… Tháng …… Năm 2024

Sinh viên

Nguyễn Thị Sương Anh

Nguyễn Thị Kiều Anh

# **DANH MỤC CÁC KÝ HIỆU, TỪ VIẾT TẮT VÀ TỪ KHOÁ**

……………………………………………………………………………………

**MỤC LỤC**

[**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN** 3](#_Toc217776592)

[**LỜI CAM ĐOAN** 4](#_Toc217776593)

[**DANH MỤC CÁC KÝ HIỆU, TỪ VIẾT TẮT VÀ TỪ KHOÁ** 5](#_Toc217776594)

[**CHƯƠNG 1: TỔNG QUAN** 10](#_Toc217776595)

[**1.1 Giới thiệu đề tài** 10](#_Toc217776596)

[**1.2 Nhiệm vụ đề tài** 11](#_Toc217776597)

[*1.2.1 Tính cấp thiết của đề tài* 11](#_Toc217776598)

[*1.2.2 Ý nghĩa khoa học và thực tiễn của đề tài* 12](#_Toc217776599)

[**1.3 Mục tiêu** 13](#_Toc217776600)

[*1.3.1 Mục tiêu tổng quát* 13](#_Toc217776601)

[**1.4 Đối tượng và phạm vi** 13](#_Toc217776602)

[*1.4.1 Đối tượng* 13](#_Toc217776603)

[*1.4.2 Phạm vi* 13](#_Toc217776604)

[**1.5 Phương pháp nghiên cứu** 14](#_Toc217776605)

[*1.5.1 Phương pháp nghiên cứu sơ bộ* 14](#_Toc217776606)

[*1.5.2 Phương pháp nghiên cứu tài liệu* 14](#_Toc217776607)

[*1.5.3 Phương pháp thực nghiệm* 14](#_Toc217776608)

[*1.5.4 Phương pháp đánh giá* 14](#_Toc217776609)

[**1.6 Những đóng góp nghiên cứu của đề tài** 15](#_Toc217776610)

[*1.6.1 Trong lĩnh vực học thuật* 15](#_Toc217776611)

[*1.6.2 Trong thực tiễn* 15](#_Toc217776612)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 15](#_Toc217776613)

[**2.1 Khái niệm cào dữ liệu (Web Scraping)** 15](#_Toc217776614)

[*2.1.1 Định nghĩa* 15](#_Toc217776615)

[*2.1.2 Cách thức hoạt động* 16](#_Toc217776616)

[*2.1.3 Các công cụ phổ biến* 16](#_Toc217776617)

[**2.2 Selenium** 16](#_Toc217776618)

[*2.2.1 Selenium là gì?* 16](#_Toc217776619)

[*2.2.2 Các thành phần chính của Selenium* 17](#_Toc217776620)

[*2.2.4 Ưu và nhược điểm* 17](#_Toc217776621)

[*2.2.5 Ứng dụng* 18](#_Toc217776622)

[**2.3 MongoDB là gì?** 19](#_Toc217776623)

[*2.3.1 Định nghĩa* 19](#_Toc217776624)

[*2.3.2 Một số đặc điểm chính của MongoDB* 19](#_Toc217776625)

[**CHƯƠNG 3: PHƯƠNG PHÁP THỰC NGHIỆM** 22](#_Toc217776626)

[**3.1 Sơ đồ mô tả quá trình thu thập dữ liệu** 22](#_Toc217776627)

[**3.2 Phương pháp thu thập dữ liệu** 23](#_Toc217776628)

[*3.2.1 Một số thư viện cần dùng* 23](#_Toc217776629)

[*3.2.2 Kiến trúc chương trình và luồng xử lý dữ liệu* 25](#_Toc217776632)

[*3.2.2.1 Giai đoạn 1 – Thu thập danh sách quán ăn* 26](#_Toc217776633)

[*3.2.2.2 Giai đoạn 2 – Thu thập thông tin đánh giá tổng hợp của quán (review\_quán)* 26](#_Toc217776634)

[*3.2.2.3 Giai đoạn 3 – Thu thập đánh giá chi tiết từ người dùng (review\_user)* 27](#_Toc217776635)

[*3.2.2.4 Tổng kết luồng xử lý dữ liệu* 28](#_Toc217776636)

[*3.2.3 Thu thập dữ liệu* 28](#_Toc217776637)

[*3.2.3.1 File 1: restaurants\_all\_districts\_from\_home\_1.py – Cào link quán từ trang chủ Foody* 28](#_Toc217776638)

[*3.2.3.2 File 2: review\_quan\_restaurants\_all.py – Cào bảng review\_quán* 37](#_Toc217776639)

[*3.2.3.3 File 3: review\_user\_all.py – Cào bảng review\_user* 43](#_Toc217776640)

[**3.3 Mô tả kết quả thu thập dữ liệu và cấu trúc dữ liệu** 48](#_Toc217776641)

[*3.3.1 Bảng 1 – Danh sách quán ăn* 48](#_Toc217776642)

[*3.3.2 Bảng 2 – review\_quán* 49](#_Toc217776643)

[*3.3.3 Bảng 3 – review\_user* 51](#_Toc217776644)

[**3.4 Phương pháp tiền xử lý dữ liệu** 52](#_Toc217776645)

[**3.5 Phương pháp phân tích dữ liệu** 55](#_Toc217776646)

[*3.5.1 Kết quả thu thập dữ liệu* 55](#_Toc217776647)

[*3.5.2 Phương pháp phân tích* 56](#_Toc217776648)

[*3.5.4 Kết luận* 63](#_Toc217776649)

[**CHƯƠNG 4: KẾT LUẬN VÀ KIẾN NGHỊ** 64](#_Toc217776650)

[**4.1 Kết luận** 64](#_Toc217776651)

[**4.2 Kiến nghị** 64](#_Toc217776652)

[**TÀI LIỆU THAM KHẢO** 66](#_Toc217776653)

MỤC LỤC HÌNH ẢNH

[Hình 3.2.1 : Hình ảnh minh họa thư viện 24](#_Toc217770367)

[Hình 3.2.2 :Cấu hình Firefox WebDriver và tham số chờ 28](#_Toc217770368)

[Hình 3.2.3 :Thao tác mở trang login và submit form đăng nhập 29](#_Toc217770369)

[Hình 3.2.4 :Vòng lặp bấm “Xem thêm” và kiểm tra dữ liệu có tăng 31](#_Toc217770370)

[Hình 3.2.5 : Hàm lấy danh sách quán 33](#_Toc217770371)

[Hình 3.2.6 : Hàm tách quận/huyện từ địa chỉ quán 34](#_Toc217770372)

[Hình 3.2.7 : Lưu dữ liệu MongoDB theo cơ chế UPSERT 35](#_Toc217770373)

[Hình 3.2.8: Xuất file Excel theo quận/huyện 36](#_Toc217770374)

[Hình 3.2.9: Đọc file Excel danh sách quán và kiểm tra dữ liệu đầu vào 37](#_Toc217770375)

[Hình 3.2.10:Khởi tạo MongoDB cho bảng review\_quán và tạo unique index chống trùng 38](#_Toc217770376)

[Hình 3.2.11: Cơ chế skip quán đã có đủ dữ liệu (tối ưu thời gian crawl) 38](#_Toc217770377)

[Hình 3.2.12: Mở trang quán, chờ tải và cuộn để hiện bảng tiêu chí 39](#_Toc217770378)

[Hình 3.2.14 : Hàm đọc bảng 5 tiêu chí và tính điểm trung bình 41](#_Toc217770379)

[Hình 3.2.15: Lưu kết quả và xuất Excel 42](#_Toc217770380)

[Hình 3.2.16 : Chuẩn bị danh sách quán trước khi cào review\_user 43](#_Toc217770381)

[Hình 3.2.17: Tạo đường dẫn trang bình luận của quán 44](#_Toc217770382)

[Hình 3.2.18 :Cơ chế bấm “Xem thêm bình luận” và kiểm tra tăng dữ liệu 44](#_Toc217770383)

[Hình 3.2.19:Trích xuất dữ liệu chính của một bình luận 45](#_Toc217770384)

[Hình 3.2.20 :Thu thập link ảnh/video trong bình luận 46](#_Toc217770385)

[Hình 3.2.21 :Tạo khóa duy nhất review\_id và lưu MongoDB 47](#_Toc217770386)

[Hình 3.2.22:Xuất file review\_user\_all.xlsx theo district 48](#_Toc217770387)

[Hình 3.4.1 : Tiền xử lý chuỗi địa chỉ trước khi trích xuất quận/huyện 52](#_Toc217770388)

[Hình 3.4.2: Chuẩn hóa dữ liệu điểm số 53](#_Toc217770389)

[Hình 3.4.3 : Quy trình chuẩn hóa và tổng hợp điểm 5 tiêu chí đánh giá 54](#_Toc217770390)

[Hình 3.4.4 : Chuẩn hóa user\_rating trong bảng review\_user 54](#_Toc217770391)

[Hình 3.5.1: Thống kê số lượng đánh giá theo từng quán 57](#_Toc217770392)

[Hình 3.5.2: Truy vấn kiểm tra tỷ lệ dữ liệu thiếu trong các trường quan trọng 58](#_Toc217770393)

[Hình 3.5.3: Kiểm tra trùng lặp 58](#_Toc217770394)

[Hình 3.5.4: Phân tích sự phân bố điểm đánh giá của người dùng 59](#_Toc217770395)

[Hình 3.5.5: Phân loại đánh giá theo mức độ hài lòng của người dùng 60](#_Toc217770396)

[Hình 3.5.6: Tìm ra thể loại quá nào có điểm trung bình tiêu chí trên 8.0 61](#_Toc217770397)

[Hình 3.5.7: Top 100 quán theo từng tiêu chí 62](#_Toc217770398)

[Hình 3.5.8: Tìm ra quán có điểm cao nhưng review thấp 63](#_Toc217770399)

[Hình 3.5.9: So sánh điểm đánh giá người dùng và điểm tiêu chí của quán ăn 63](#_Toc217770400)

[Hình 3.5.10: Quán đáng tin để đề xuất 64](#_Toc217770401)

# **CHƯƠNG 1: TỔNG QUAN**

## **1.1 Giới thiệu đề tài**

Trong bối cảnh phát triển của công nghệ và thương mại điện tử, việc lựa chọn quán ăn không chỉ dựa vào trải nghiệm cá nhân mà còn chịu ảnh hưởng từ các nền tảng đánh giá trực tuyến. Foody.vn là một nền tảng phổ biến tại Việt Nam, nơi mà người dùng chia sẻ cảm nhận, đánh giá và hình ảnh về các quán ăn.Đây là một nguồn dữ liệu khổng lồ nếu được khai thác một cách hợp lý sẽ mang lại giá trị lớn trong việc hỗ trợ người dùng đưa ra quyết định lựa chọn quán ăn phù hợp.

Tuy nhiên, việc khai thác và phân tích hiệu quả nguồn dữ liệu này cần sử dụng các công cụ lập trình và kỹ thuật thu thập dữ liệu phù hợp. Do đó, đề tài “Phân tích và đề xuất quán ăn dựa trên dữ liệu đánh giá người dùng từ Foody.vn sử dụng công cụ Selenium và MongoDB” được thực hiện nhằm xây dựng quy trình thu thập dữ liệu tự động, lưu trữ và phân tích đánh giá của người dùng, từ đó hỗ trợ việc đề xuất quán ăn một cách khách quan và hiệu quả.

## **1.2 Nhiệm vụ đề tài**

Nhiệm vụ của đề tài “Phân tích và đề xuất quán ăn dựa trên dữ liệu đánh giá người dùng từ Foody.vn sử dụng công cụ Selenium và MongoDB” là nghiên cứu việc sử dụng Selenium để tự động hóa quá trình thu thập dữ liệu từ Foody.vn , đồng thời lựa chọn MongoDB để lưu trữ và quản lý dữ liệu một cách linh hoạt. Trên cơ sở dữ liệu thu thập được, tiến hành phân tích các đánh giá của người dùng, từ đó xây dựng cơ sở đề xuất quán ăn và đánh giá hiệu quả của phương pháp thực hiện.

## *1.2.1 Tính cấp thiết của đề tài*

Trong bối cảnh công nghệ thông tin và thương mại điện tử phát triển mạnh mẽ, nhu cầu lựa chọn quán ăn của người dùng ngày càng đa dạng và phức tạp. Người tiêu dùng không chỉ quan tâm đến chất lượng món ăn, mà còn quan tâm đến giá cả, không gian và dịch vụ. Tuy nhiên, việc đưa ra quyết định ngày càng khó khăn vì số lượng quán ăn lớn và thông tin phân tán. Foody.vn là một nền tảng phổ biến, là nơi tập trung nguồn dữ liệu đánh giá khổng lồ từ người dùng thực tế. Việc khai thác và phân tích nguồn dữ liệu này là hợp lý để cung cấp một cái nhìn khách quan(đa chiều), hỗ trợ người dùng đưa ra quyết định nhanh chóng và chính xác hơn.  
Selenium là một công cụ mạnh trong lập trình Python, được phát triển nhằm hỗ trợ tự động hóa trình duyệt và xử lý các nội dung web động, đóng vai trò quan trọng trong quá trình thu thập dữ liệu web (web scraping), đặc biệt với những nền tảng phức tạp như Foody.vn - nơi dữ liệu được tải tự động và khó truy xuất bằng phương pháp truyền thống. Khi sử dụng Selenium để thu thập dữ liệu đánh giá quán ăn, người dùng có thể dễ dàng truy xuất các thông tin như tên quán, địa chỉ, điểm số đánh giá, nội dùng reviews. Điều này giúp người người đưa ra lựa chọn quán ăn phù hợp.

Tuy nhiên, dữ liệu thu thập được đưa ra ở dạng không có cấu trúc cố định, đòi hỏi hệ quản trị cơ sở dữ liệu linh hoạt để lưu trữ và xử lý. MongoDB được lựa chọn như một giải pháp phù hợp, cho phép quản lý dữ liệu dạng văn bản như các bài đánh giá của người dùng, đồng thời hỗ trợ truy vấn nhanh và khả năng mở rộng dễ dàng.

Tóm lại, việc nghiên cứu và ứng dụng Selenium kết hợp với MongoDB trong thu thập và phân tích dữ liệu từ Foody.vn là một đề tài mang tính cấp thiết, giúp người dùng có công cụ tham khảo khách quan khi lựa chọn món ăn, hỗ trợ chủ quán cải thiện chất lượng dịch vụ dựa trên đánh giá thực tế. Đồng thời, đề tài còn làm nổi bật vai trò của công nghệ dữ liệu lớn trong việc cải thiện trải nghiệm ẩm thực.

## *1.2.2 Ý nghĩa khoa học và thực tiễn của đề tài*

Ý nghĩa khoa học: Đề tài cung cấp một nghiên cứu thực nghiệm về việc ứng dụng các công cụ hiện đại trong lĩnh vực khai thác dữ liệu web. Việc sử dụng Selenium giúp tự động hóa quá trình thu thập dữ liệu từ các trang web động như Foody.vn , nơi thông tin được đăng tải liên tục và khó truy xuất bằng cái phương pháp truyền thống. Đồng thời, MongoDB hỗ trợ lưu trữ và quản lý dữ liệu không có cấu trúc cố định như nội dung reviews, tạo điều kiện thuận lợi cho việc phân tích dữ liệu lớn, khai phá thông tin. Đây là cơ sở để mở rộng nghiên cứu về phân tích dữ liệu trực tuyến, xử lý ngôn ngữ tự nhiên và ứng dụng trí tuệ nhân tạo trong lĩnh vực ăn uống.

Ý nghĩa thực tiễn: Về mặt thực tiễn, đề tài mang lại giá trị rõ rệt khi giúp người dùng có công cụ tham khảo khách quan để lựa chọn quán ăn phù hợp với nhu cầu, tiết kiệm thời gian và nâng cao trải nghiệm ẩm thực. Kết quả phân tích dữ liệu đánh giá cũng hỗ trợ chủ quán hiểu rõ hơn về phản hồi của khách hàng, từ đó cải thiện chất lượng món ăn, dịch vụ và không gian. Ngoài ra, đề tài còn cung cấp cái nhìn tổng quan về xu hướng tiêu dùng trong ngành ẩm thực, thể hiện tính ứng cao của công nghệ dữ liệu lớn trong đời sống, đặc biệt trong lĩnh vực dịch vụ ăn uống và thương mại điện tử.

## **1.3 Mục tiêu**

### *1.3.1 Mục tiêu tổng quát*

Đề tài “Phân tích và đề xuất quán ăn dựa trên dữ liệu đánh giá người dùng từ Foody.vn sử dụng công cụ Selenium và MongoDB” nhằm nghiên cứu và ứng dụng Selenium để thu thập dữ liệu đánh giá từ Foody.vn một cách hiệu quả, kết hợp với MongoDB để lưu trữ và quản lý dữ liệu. Trên cơ sở dữ liệu thu thập được, tiến hành phân tích các đánh giá của người dùng như bình luận, hình ảnh, điểm trung bình, từ đó hỗ trợ việc đề xuất quán ăn phù hợp, phục vụ người dùng

1.3.2 Mục tiêu cụ thể

Mục tiêu cụ thể của đề tài *“Phân tích và đề xuất quán ăn dựa trên dữ liệu đánh giá người dùng từ Foody.vn sử dụng công cụ Selenium và MongoDB”* là xây dựng quy trình thu thập dữ liệu đánh giá từ Foody.vn bằng Selenium, bao gồm các thông tin như điểm số, bình luận, hình ảnh và thời gian đánh giá, đồng thời lưu trữ và quản lý dữ liệu này trong MongoDB một cách linh hoạt và hiệu quả. Trên cơ sở dữ liệu thu thập được, đề tài tiến hành tiền xử lý và phân tích các đánh giá của người dùng nhằm xác định mức độ hài lòng và các yếu tố ảnh hưởng đến lựa chọn quán ăn. Kết quả phân tích được sử dụng để đề xuất các quán ăn phù hợp, hỗ trợ người dùng ra quyết định khách quan.

## **1.4 Đối tượng và phạm vi**

### *1.4.1 Đối tượng*

Đối tượng nghiên cứu của đề tài là **dữ liệu đánh giá của người dùng trên nền tảng Foody.vn**, bao gồm các thông tin như điểm đánh giá, nội dung bình luận, hình ảnh, thời gian đăng và mức độ tương tác của người dùng đối với các quán ăn.

### *1.4.2 Phạm vi*

Phạm vi nghiên cứu bao gồm việc thu thập dữ liệu từ các quán ăn tại Thành phố Hồ Chí Minh, tập trung vào những quán ăn phổ biến và có lượng đánh giá lớn. Dữ liệu nghiên cứu bao gồm các bình luận, điểm đánh giá, hình ảnh và thời gian đánh giá liên quan đến các quán ăn, không bao gồm thông tin cá nhân ngoài những dữ liệu công khai trên website. Dữ liệu được thu thập trong một khoảng thời gian xác định, đảm bảo tính đại diện và đầy đủ. Về mặt kỹ thuật, đề tài sử dụng Selenium để thu thập dữ liệu tự động và MongoDB để lưu trữ, quản lý, xử lý và phân tích dữ liệu, nhằm phục vụ cho việc đề xuất quán ăn và đánh giá hiệu quả của phương pháp nghiên cứu.

## **1.5 Phương pháp nghiên cứu**

### *1.5.1 Phương pháp nghiên cứu sơ bộ*

Phương pháp nghiên cứu sơ bộ được thực hiện nhằm khảo sát tổng quan về nền tảng Foody.vn , xác định các loại dữ liệu đánh giá người dùng, cũng như tìm hiểu các công cụ và kỹ thuật thu thập dữ liệu web hiện có. Thông qua việc nghiên cứu sơ bộ, nhóm xác định được các yêu cầu kỹ thuật và phạm vi dữ liệu phù hợp cho đề tài.

### *1.5.2 Phương pháp nghiên cứu tài liệu*

Chúng tôi tiến hành thu thập và nghiên cứu các tài liệu, bài báo, sách chuyên khảo liên quan đến web scraping, cơ sở dữ liệu NoSQL (MongoDB), phân tích dữ liệu người dùng và các hệ thống đề xuất. Việc nghiên cứu tài liệu giúp làm cơ sở lý thuyết, xây dựng quy trình nghiên cứu và lựa chọn công cụ phù hợp cho việc thu thập và phân tích dữ liệu.

### *1.5.3 Phương pháp thực nghiệm*

Phương pháp thực nghiệm được áp dụng thông qua việc lập trình thu thập dữ liệu từ Foody.vn bằng Selenium, lưu trữ dữ liệu trong MongoDB, và tiến hành tiền xử lý, phân tích dữ liệu đánh giá. Quá trình thực nghiệm bao gồm kiểm tra tính đầy đủ, chính xác của dữ liệu, đồng thời đánh giá khả năng mở rộng và hiệu quả của hệ thống thu thập dữ liệu.

### *1.5.4 Phương pháp đánh giá*

Phương pháp đánh giá được thực hiện dựa trên việc phân tích kết quả dữ liệu thu thập và xử lý, đánh giá chất lượng, tính đại diện và tính chính xác của thông tin. Kết quả phân tích được so sánh với các tiêu chí thực tế để xác định hiệu quả của phương pháp thu thập và phân tích dữ liệu, đồng thời đánh giá khả năng hỗ trợ người dùng trong việc đề xuất quán ăn phù hợp.

## **1.6 Những đóng góp nghiên cứu của đề tài**

### *1.6.1 Trong lĩnh vực học thuật*

Đề tài cung cấp một quy trình nghiên cứu hoàn chỉnh về việc thu thập và phân tích dữ liệu đánh giá người dùng từ nền tảng trực tuyến sử dụng Selenium và MongoDB, góp phần làm cơ sở cho các nghiên cứu tiếp theo về phân tích dữ liệu người dùng và hệ thống đề xuất.

### *1.6.2 Trong thực tiễn*

Hỗ trợ người dùng ra quyết định khách quan khi lựa chọn quán ăn, giúp chủ quán cải thiện chất lượng dịch vụ dựa trên phản hồi thực tế, đồng thời minh họa ứng dụng công nghệ dữ liệu lớn trong lĩnh vực ẩm thực và nghiên cứu thị trường.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1 Khái niệm cào dữ liệu (Web Scraping)**

### *2.1.1 Định nghĩa*

**Web Scraping** là một công nghệ cho phép tự động thu thập dữ liệu từ các trang web trên internet. Quá trình này thực hiện thông qua việc lấy các thông tin trên trang web dựa trên cú pháp HTML của trang đó. Sau đó, thông tin được lưu trữ trong một cơ sở dữ liệu hoặc một tệp tin để sử dụng cho mục đích nghiên cứu, phân tích hoặc báo cáo.

### *2.1.2 Cách thức hoạt động*

Quá trình cào dữ liệu thường diễn ra theo các bước sau đây:

* Gửi yêu cầu HTML
* Phân tích cú pháp(parsing)
* Trích xuất dữ liệu
* Lưu trữ dữ liệu

### *2.1.3 Các công cụ phổ biến*

Một số công cụ phổ biến trong web scraping bao gồm:

* BeautifulSoup
* Scrapy
* Selenium
* Requests

## **2.2 Selenium**

### *2.2.1 Selenium là gì?*

Selenium là một công cụ mã nguồn mở dùng để tự động hóa trình duyệt web.Nó cho phép lập trình viên mô phỏng các thao tác của người dùng như click, scroll, nhập liệu, chọn tab hay tải dữ liệu động, và nó chủ yếu được sử dụng cho việc kiểm thử (testing). Selenium hỗ trợ nhiều ngôn ngữ lập trình như Python, Java, C#, giúp tích hợp linh hoạt với các thư viện phân tích dữ liệu và cơ sở dữ liệu.

### *2.2.2 Các thành phần chính của Selenium*

Selenium Webdriver: là một automation testing tool dành riêng cho web, nó sẽ gửi lệnh khởi chạy và tương tác trực tiếp tới các trình duyệt.

Selenium IDE: Selenium Integrated Development Environment (IDE) là một plugin trên trình duyệt Chrome và Firefox. Ta có thể sử dụng chúng để ghi và phát lại(record and playback) các tương tác của người dùng theo một quy trình hay một test case nào đó.

Selenium Grid: được sử dụng để khởi chạy nhiều kịch bản test song song cùng một lúc, và có thể chạy trên nhiều máy, nhiều hệ điều hành và nhiều trình duyệt khác nhau.

Selenium RC: Selenium Remote Control (RC), Selenium server khởi chạy và điều khiển trình duyệt web từ xa, hỗ trợ kiểm thử ứng dụng web trên nhiều trình duyệt và hệ điều hành.

### *2.2.4 Ưu và nhược điểm*

Ưu điểm

* Mã nguồn mở/Không mất phí bản quyền: Selenium không yêu cầu bất kỳ chi phí trả trước hay thanh toán nào để sử dụng. Nó là phần mềm mã nguồn mở, cho phép mở rộng và sửa đổi khung nền tảng cơ bản để sử dụng cho mục đích cá nhân hoặc chuyên nghiệp
* Tích hợp với nhiều ngôn ngữ lập trình: Selenium cung cấp các thư viện chính thứ cho C#, Java, Ruby, JavaScript, PHP, Python. Ngoài ra còn một số thư viện không chính thức cho các ngôn ngữ còn lại
* Hỗ trợ đa trình duyệt**:** Selenium hoạt động trên nhiều trình duyệt phổ biến như Chrome, Firefox, Edge, Safari, giúp linh hoạt trong quá trình thực nghiệm.
* Tự động hóa trình duyệt**:** Selenium có thể mô phỏng các thao tác của người dùng như click, scroll, điền form, giúp thu thập dữ liệu từ các trang web động mà các phương pháp thu thập dữ liệu thông thường không làm được.
* Mở rộng và tích hợp**:** Selenium có thể kết hợp với các công cụ khác như BeautifulSoup, Pandas, MongoDB để thu thập, lưu trữ và phân tích dữ liệu một cách đồng bộ.

Nhược điểm

* Chỉ hỗ trợ ứng dụng Web: Selenium không thể tự động hóa kiểm thử cho các ứng dụng Desktop (ứng dụng trên máy tính) hoặc Mobile (di động) một cách trực tiếp, chỉ giới hạn trong môi trường trình duyệt web.
* Thiếu tính năng báo cáo tích hợp: Selenium không có sẵn chức năng tạo báo cáo tự động, phải kết hợp với các framework bên thứ ba như JUnit, TestNG, hoặc Cucumber.
* Khó xử lý các thành phần phức tạp:Gặp khó khăn khi xử lý các phần tử động (dynamic elements), cửa sổ pop-up, khung (frames).Không hỗ trợ tự động hóa xác thực Captcha/Mã vạch (Barcode).

### *2.2.5 Ứng dụng*

Selenium là một công cụ mạnh mẽ trong Python, được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau. Trước hết, Selenium thường được sử dụng trong kiểm thử phần mềm tự động (automation testing), giúp mô phỏng thao tác của người dùng trên trình duyệt để kiểm tra giao diện, chức năng và khả năng tương thích của ứng dụng web trên nhiều trình duyệt khác nhau. Bên cạnh đó, Selenium còn được ứng dụng trong thu thập dữ liệu web (web scraping), đặc biệt hiệu quả với các trang web động như Foody.vn hay Shopee.vn, nơi nội dung được tải bằng JavaScript và khó truy xuất bằng các thư viện xử lý HTML tĩnh. Ngoài ra, Selenium có thể tích hợp với các công cụ khác như BeautifulSoup để phân tích HTML, MongoDB để lưu trữ dữ liệu, hoặc các framework kiểm thử như JUnit, TestNG, PyTest để quản lý và báo cáo kết quả kiểm thử. Nhờ khả năng tự động hóa linh hoạt, Selenium trở thành công cụ quan trọng trong cả nghiên cứu dữ liệu lẫn phát triển phần mềm, giúp tiết kiệm thời gian, nâng cao hiệu quả và mở rộng phạm vi ứng dụng công nghệ trong thực tiễn.

## **2.3 MongoDB là gì?**

### *2.3.1 Định nghĩa*

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL mã nguồn mở, được thiết kế để lưu trữ và quản lý dữ liệu phi cấu trúc hoặc bán cấu trúc. Khác với các cơ sở dữ liệu quan hệ truyền thống (SQL) sử dụng bảng và dòng, MongoDB lưu trữ dữ liệu dưới dạng document theo chuẩn BSON (Binary JSON). Mỗi document tương tự một bản ghi JSON, có thể chứa nhiều trường dữ liệu khác nhau như văn bản, số, danh sách hoặc nhúng các document khác, giúp dữ liệu linh hoạt và dễ mở rộng.

### *2.3.2 Một số đặc điểm chính của MongoDB*

* **Lưu trữ dữ liệu dạng document:** Dữ liệu được lưu dưới dạng BSON (Binary JSON), cho phép lưu trữ văn bản, số, danh sách, mảng và các document lồng nhau.
* **Schema linh hoạt:** Không cần định nghĩa schema cố định, các document trong cùng một collection có thể khác nhau về cấu trúc.
* **Khả năng mở rộng cao:** Hỗ trợ sharding và replication, giúp xử lý dữ liệu lớn và phân tán trên nhiều server.
* **Truy vấn mạnh mẽ:** Hỗ trợ truy vấn phức tạp, tìm kiếm văn bản, lọc theo trường, sắp xếp và thống kê.
* **Tích hợp dễ dàng với các ngôn ngữ lập trình:** Hỗ trợ Python, Java, C#, Node.js…, giúp kết hợp trực tiếp với các công cụ phân tích dữ liệu.
* **Hiệu năng cao với dữ liệu phi cấu trúc:** Thích hợp cho các ứng dụng cần lưu trữ dữ liệu thay đổi thường xuyên hoặc dữ liệu không đồng nhất.

2.3.3 Ưu và nhược điểm

Ưu điểm

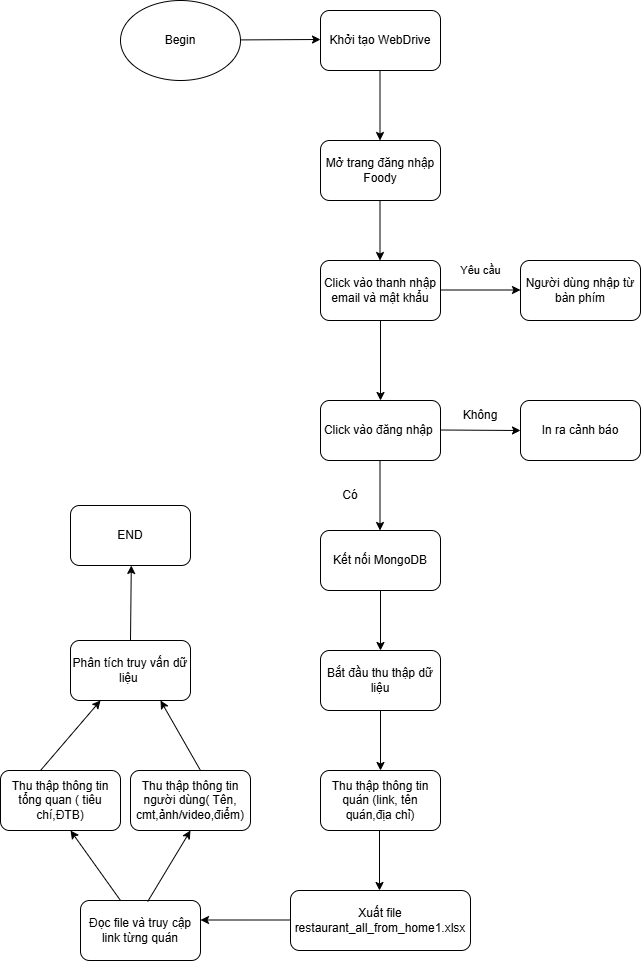
* **Linh hoạt:** Dữ liệu phi cấu trúc hoặc bán cấu trúc đều có thể lưu trữ dễ dàng.
* **Mở rộng dễ dàng:** Hỗ trợ sharding và replication cho dữ liệu lớn.
* **Truy vấn nhanh và mạnh mẽ:** Hỗ trợ các truy vấn phức tạp, thống kê và tìm kiếm văn bản.
* **Tích hợp tốt với Python và các thư viện phân tích dữ liệu:** Giúp đồng bộ giữa thu thập, lưu trữ và xử lý dữ liệu.
* **Không cần schema cố định:** Phù hợp với dữ liệu web biến đổi hoặc dữ liệu thử nghiệm.

Nhược điểm

* **Tiêu tốn bộ nhớ:** Dữ liệu dạng BSON có thể chiếm nhiều dung lượng hơn so với cơ sở dữ liệu quan hệ.
* **Xử lý join hạn chế:** Khi dữ liệu liên quan nhiều collection, việc join phức tạp và kém hiệu quả hơn cơ sở dữ liệu SQL.
* **Phụ thuộc vào lập trình viên:** Cần thiết kế schema, index và truy vấn hiệu quả để tối ưu hóa hiệu năng.
* **Không phù hợp với dữ liệu quá nhỏ:** Dự án đơn giản hoặc dữ liệu ít có thể không cần dùng MongoDB, vì sẽ phức tạp và thừa tính năng.
* **Yêu cầu bảo mật và quản lý tốt:** Cần xác thực, phân quyền và sao lưu để tránh mất mát hoặc truy cập trái phép.

# **CHƯƠNG 3: PHƯƠNG PHÁP THỰC NGHIỆM**

## **3.1 Sơ đồ mô tả quá trình thu thập dữ liệu**

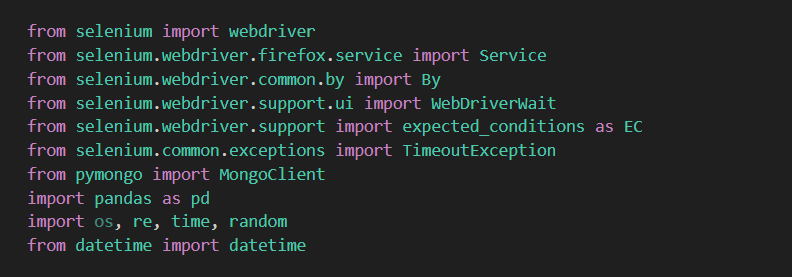


Hình 3.1.1: Sơ dồ mô tả quá trình thu thập dữ liệu

## **3.2 Phương pháp thu thập dữ liệu**

### *3.2.1 Một số thư viện cần dùng*

**



### 

### 

Hình 3.2.1 : Hình ảnh minh họa thư viện

* **selenium (webdriver):** Thư viện chính để tự động hóa trình duyệt (mở trang, click, nhập dữ liệu, cuộn trang…) nhằm thu thập dữ liệu từ Foody một cách tự động.
* **selenium.webdriver.firefox.service.Service:** Dùng để khởi tạo dịch vụ Firefox Driver (geckodriver), giúp Selenium điều khiển trình duyệt Firefox ổn định theo đúng đường dẫn driver.
* **selenium.webdriver.common.by.By**: Cung cấp các cách định vị phần tử HTML (theo ID, CLASS, CSS selector, XPath, …) để tìm đúng nút/khối dữ liệu cần crawl.
* **selenium.webdriver.support.ui.WebDriverWait:** Hỗ trợ chờ phần tử xuất hiện/khả dụng theo thời gian tối đa (timeout), giúp hạn chế lỗi do mạng chậm hoặc trang tải chưa kịp.
* **selenium.webdriver.support.expected\_conditions as EC:** Tập điều kiện chờ (ví dụ: phần tử xuất hiện, có thể click…) kết hợp với WebDriverWait để đợi đúng trạng thái trước khi thao tác.
* **selenium.common.exceptions.TimeoutException:** Bắt lỗi quá thời gian chờ (khi phần tử không xuất hiện đúng thời gian), giúp chương trình không bị dừng đột ngột và có thể xử lý/skip hợp lý.
* **time:** Dùng để tạm dừng (sleep) hoặc hỗ trợ chờ thêm trong một số tình huống tải trang/scroll/load bình luận.
* **re:** Thư viện Regular Expression để xử lý chuỗi, ví dụ: làm sạch text, tách thông tin trong địa chỉ, chuẩn hóa dữ liệu dạng ký tự.
* **hashlib:** Dùng để tạo mã băm (hash). Trong bài toán crawl review, có thể dùng để tạo review\_id dự phòng (khi trang không có ID sẵn), giúp đảm bảo tính duy nhất khi lưu MongoDB.
* **urllib.parse.urljoin:** Dùng để ghép URL tương đối thành URL đầy đủ, rất hữu ích khi lấy link ảnh/video dạng tương đối từ HTML.
* **pandas as pd:** Thư viện xử lý dữ liệu dạng bảng (DataFrame). Dùng để đọc Excel đầu vào, tổng hợp dữ liệu và xuất Excel theo quận/huyện.
* **pymongo.MongoClient:** Thư viện kết nối MongoDB trong Python, dùng để kết nối database, tạo index, và lưu dữ liệu theo cơ chế upsert (hạn chế trùng lặp).
* **datetime (from datetime import datetime):** Dùng để lấy thời gian hiện tại, thường dùng để tạo trường **scraped\_at** (thời điểm crawl dữ liệu), phục vụ theo dõi và kiểm tra dữ liệu.
* **os:** Thư viện thao tác hệ thống/tệp, dùng để tạo thư mục, quản lý đường dẫn file khi đọc/ghi dữ liệu (Excel/CSV…).
* **random:** Dùng để tạo độ trễ ngẫu nhiên hoặc hỗ trợ một số thao tác phụ (nếu script có áp dụng), giúp quá trình crawl tự nhiên hơn.
* **math:** Thư viện toán học cơ bản, thường dùng cho các phép tính/biến đổi số.

### *3.2.2 Kiến trúc chương trình và luồng xử lý dữ liệu*

* Chương trình thu thập dữ liệu Foody của nhóm được xây dựng theo mô hình pipeline nhiều giai đoạn, trong đó dữ liệu được thu thập và mở rộng dần qua từng bước xử lý. Thay vì thực hiện toàn bộ trong một file duy nhất, nhóm chia chương trình thành ba file Python độc lập, tương ứng với ba giai đoạn thu thập dữ liệu khác nhau. Cách tổ chức này giúp chương trình dễ quản lý, dễ mở rộng và thuận tiện cho việc kiểm tra, phân tích dữ liệu về sau.
* Luồng xử lý dữ liệu tổng quát của chương trình gồm ba giai đoạn chính:
  + Thu thập danh sách quán ăn từ trang chủ Foody
  + Từ danh sách quán, thu thập thông tin đánh giá tổng hợp của quán
  + Từ danh sách quán, thu thập chi tiết đánh giá của người dùng
* Dữ liệu ở mỗi giai đoạn đều được lưu vào MongoDB theo cơ chế chống trùng lặp, đồng thời xuất ra file Excel để phục vụ kiểm tra và phân tích.

#### *3.2.2.1 Giai đoạn 1 – Thu thập danh sách quán ăn*

* File code sử dụng: ***restaurants\_all\_districts\_from\_home\_1.py***
* Ở giai đoạn đầu tiên, chương trình thực hiện đăng nhập vào hệ thống Foody và thu thập danh sách các quán ăn hiển thị trên trang chủ. Do Foody sử dụng cơ chế tải dữ liệu động, chương trình phải mô phỏng thao tác người dùng bằng Selenium, bao gồm cuộn trang và bấm nút “Xem thêm” nhiều lần để tải đầy đủ danh sách quán.
* Mỗi quán ăn được trích xuất các thông tin cơ bản gồm: đường dẫn quán (***restaurant\_url***), tên quán (***restaurant\_name***) và địa chỉ. Từ địa chỉ, chương trình tiếp tục xử lý để xác định quận/huyện (district), phục vụ cho việc phân nhóm dữ liệu.
* Để tránh trùng lặp dữ liệu khi chương trình được chạy nhiều lần, dữ liệu quán ăn được lưu vào MongoDB theo cơ chế UPSERT, trong đó ***restaurant\_url*** đóng vai trò khóa duy nhất. Nhờ đó, nếu quán đã tồn tại trong cơ sở dữ liệu thì thông tin sẽ được cập nhật, còn nếu chưa có thì sẽ được thêm mới. Sau khi hoàn tất thu thập, toàn bộ dữ liệu được xuất ra file Excel, bao gồm một sheet tổng hợp và các sheet riêng theo từng quận/huyện.

#### *3.2.2.2 Giai đoạn 2 – Thu thập thông tin đánh giá tổng hợp của quán (review\_quán)*

* File code sử dụng: ***review\_quan\_restaurants\_all.py***
* Ở giai đoạn thứ hai, chương trình sử dụng danh sách quán đã thu thập ở giai đoạn 1 (file Excel) làm dữ liệu đầu vào. Với mỗi quán, chương trình truy cập trực tiếp vào trang chi tiết của quán để thu thập các thông tin đánh giá tổng hợp.
* Cụ thể, chương trình trích xuất thể loại quán và 5 tiêu chí đánh giá chính, bao gồm: vị trí, giá cả, chất lượng, phục vụ và không gian. Nếu thu thập đủ cả năm tiêu chí, chương trình sẽ tính toán thêm điểm trung bình của các tiêu chí này để làm đại diện cho mức đánh giá tổng thể của quán.
* Để tối ưu thời gian chạy và hạn chế việc crawl lặp lại không cần thiết, trước khi truy cập vào trang quán, chương trình kiểm tra xem quán đó đã có đầy đủ dữ liệu đánh giá trong MongoDB hay chưa. Nếu đã có đầy đủ thông tin, quán sẽ được bỏ qua. Ngược lại, chương trình mới tiến hành crawl và cập nhật dữ liệu. Kết quả cuối cùng được lưu vào MongoDB và xuất ra file Excel, được tổ chức theo sheet tổng hợp và các sheet theo quận/huyện.

#### *3.2.2.3 Giai đoạn 3 – Thu thập đánh giá chi tiết từ người dùng (review\_user)*

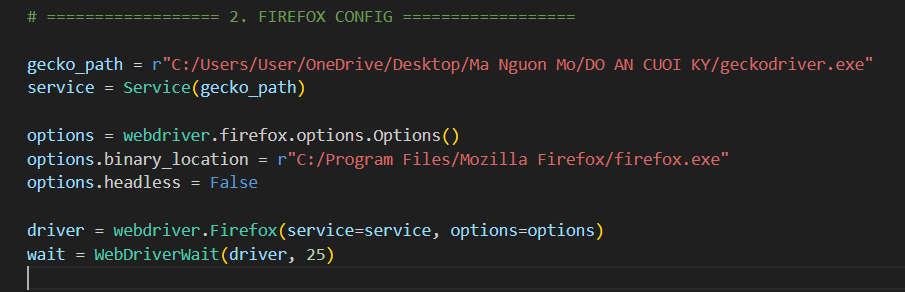
* File code sử dụng: ***review\_user\_all.py***
* Giai đoạn cuối cùng tập trung vào việc thu thập các đánh giá chi tiết của người dùng đối với từng quán ăn. Từ danh sách quán đầu vào, chương trình truy cập vào trang bình luận của quán (đường dẫn /binh-luan) để thu thập dữ liệu.
* Do số lượng bình luận có thể rất lớn và được tải động, chương trình thực hiện thao tác bấm nút “Xem thêm bình luận” nhiều lần cho đến khi không còn bình luận mới được tải thêm. Với mỗi bình luận, các thông tin chính được thu thập bao gồm: tên người dùng, điểm đánh giá, nội dung bình luận, thời gian đăng và các đường dẫn ảnh hoặc video (nếu có).
* Trong trường hợp trang không cung cấp sẵn mã định danh cho bình luận, chương trình tạo một mã ***review\_id*** dự phòng bằng cách băm (hash) nội dung bình luận và thông tin liên quan. Cách làm này giúp đảm bảo mỗi bình luận là duy nhất khi lưu trữ trong MongoDB, đồng thời tránh trùng lặp dữ liệu khi chạy lại chương trình.
* Sau khi hoàn tất thu thập, dữ liệu đánh giá người dùng được lưu vào MongoDB theo cơ chế upsert và xuất ra file Excel để phục vụ cho quá trình phân tích dữ liệu và trình bày kết quả trong báo cáo.

#### *3.2.2.4 Tổng kết luồng xử lý dữ liệu*

* Tổng thể, chương trình được thiết kế với luồng xử lý dữ liệu rõ ràng, tuần tự và nhất quán giữa các giai đoạn. MongoDB đóng vai trò là kho lưu trữ trung tâm, giúp quản lý dữ liệu hiệu quả và hạn chế trùng lặp. Các file Excel được sử dụng như đầu ra trung gian nhằm hỗ trợ kiểm tra dữ liệu và phục vụ cho bước phân tích, trực quan hóa ở các phần tiếp theo của báo cáo.

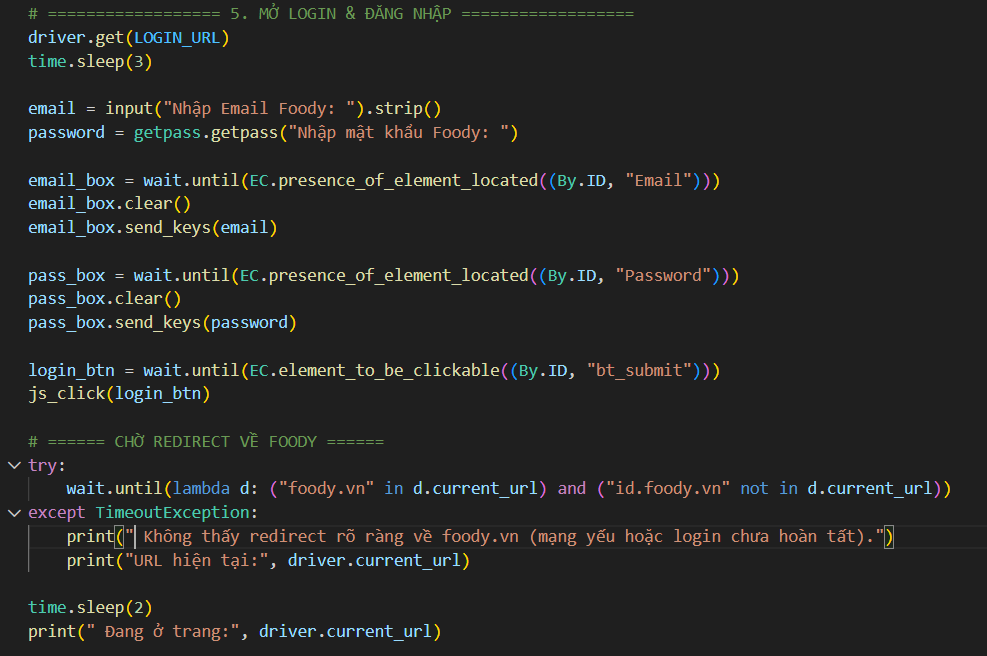
### *3.2.3 Thu thập dữ liệu*

#### *3.2.3.1 File 1: restaurants\_all\_districts\_from\_home\_1.py – Cào link quán từ trang chủ Foody*



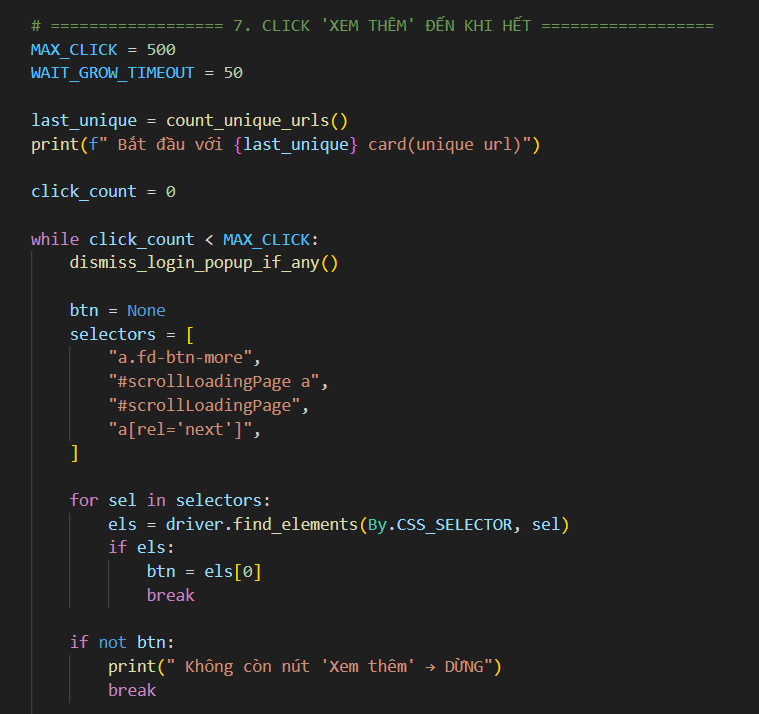
Hình 3.2.2 :Cấu hình Firefox WebDriver và tham số chờ

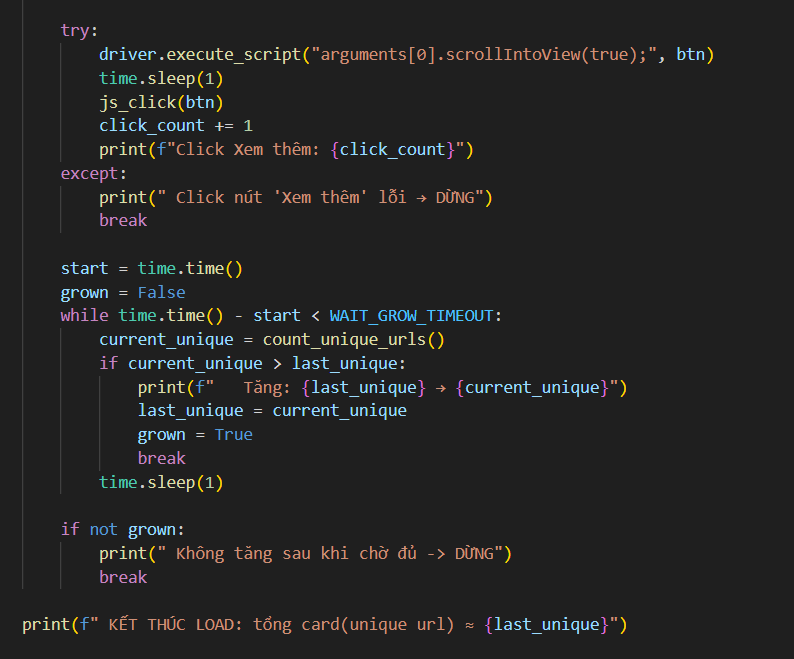
* Khai báo đường dẫn GECKO\_PATH và FIREFOX\_BINARY để Selenium điều khiển Firefox đúng môi trường.
* Thiết lập HEADLESS (ẩn/hiện trình duyệt) tùy nhu cầu chạy thử hay chạy thật.
* Dùng WAIT\_SEC và WebDriverWait để hạn chế lỗi do trang tải chậm.

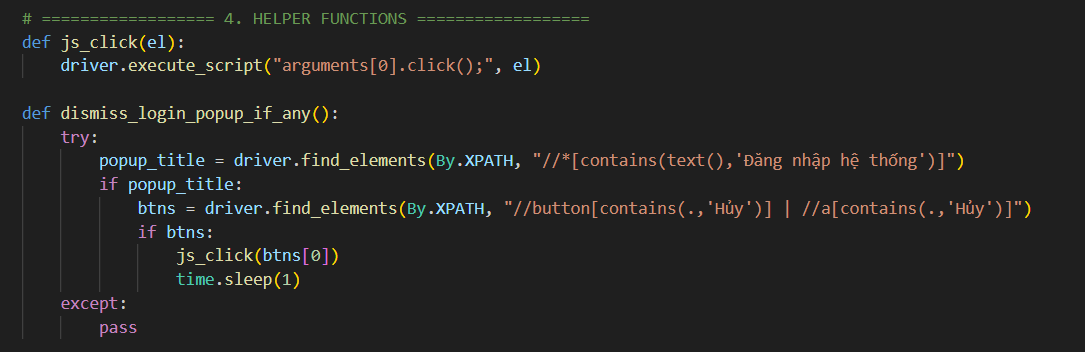


Hình 3.2.3 :Thao tác mở trang login và submit form đăng nhập

* Mở trang đăng nhập theo LOGIN\_URL.
* Tìm các ô nhập theo ID (ví dụ Email, Password) rồi send\_keys() để nhập thông tin.
* Click nút đăng nhập (ví dụ bt\_submit) để gửi form.
* Sau khi click, chương trình có thể kiểm tra redirect về domain Foody để xác nhận đăng nhập thành công.

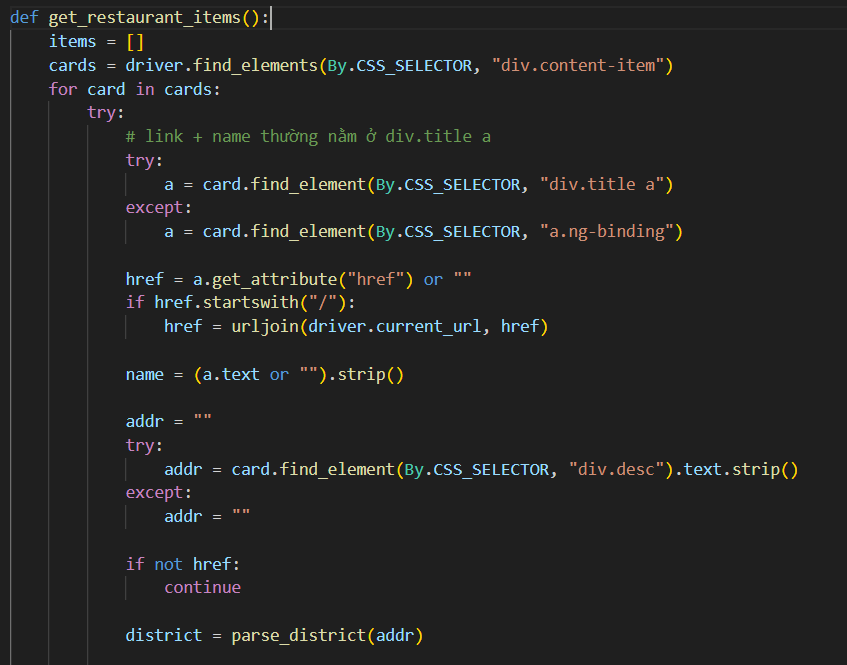






Hình 3.2.4 :Vòng lặp bấm “Xem thêm” và kiểm tra dữ liệu có tăng

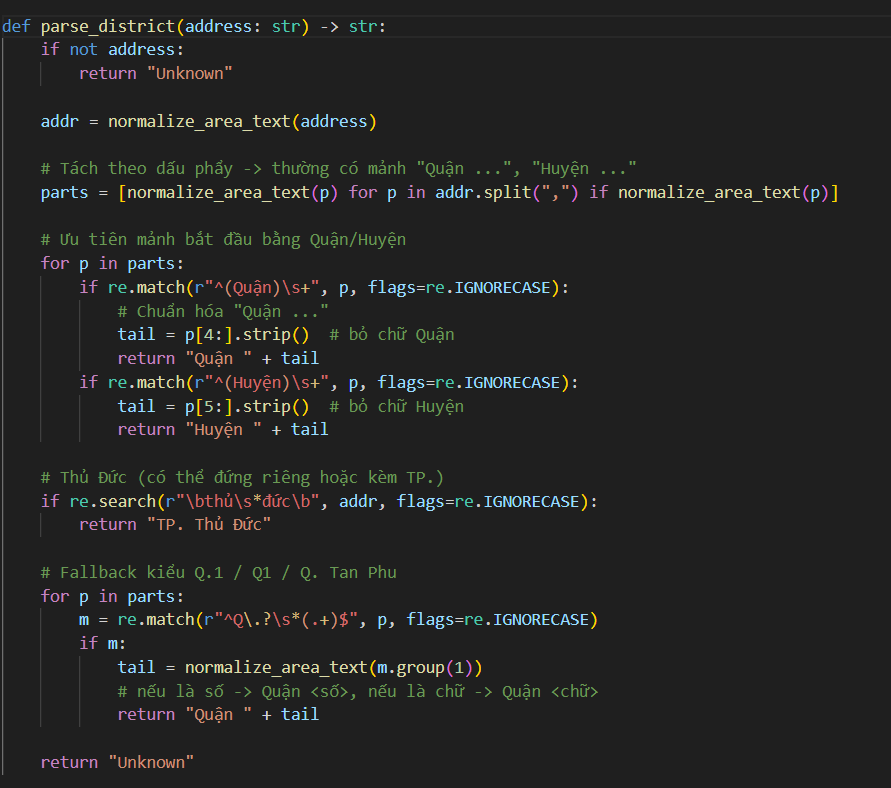
* Chương trình lặp tối đa MAX\_CLICK lần để tránh chạy vô hạn.
* Mỗi lượt:
  + Tìm nút “Xem thêm” theo nhiều selector dự phòng (đề phòng Foody thay đổi HTML).
  + Scroll tới nút rồi click để tải thêm quán.
* Sau khi click, chương trình chờ và kiểm tra số lượng restaurant\_url unique có tăng:
  + Nếu tăng → tiếp tục vòng lặp.
  + Nếu không tăng → dừng (có thể do hết dữ liệu hoặc mạng yếu).
* dismiss\_login\_popup\_if\_any() dò popup “Đăng nhập hệ thống”, nếu có thì tìm nút “Hủy” và tắt.
* js\_click() dùng execute\_script để click khi .click() thường bị lỗi do element bị che / không tương tác được.
* Giúp vòng lặp load-more chạy ổn định hơn.





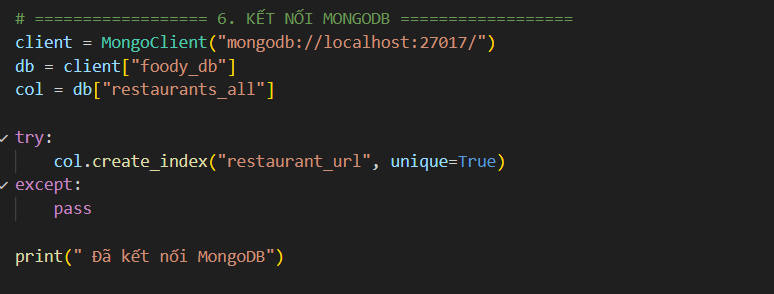
Hình 3.2.5 : Hàm lấy danh sách quán

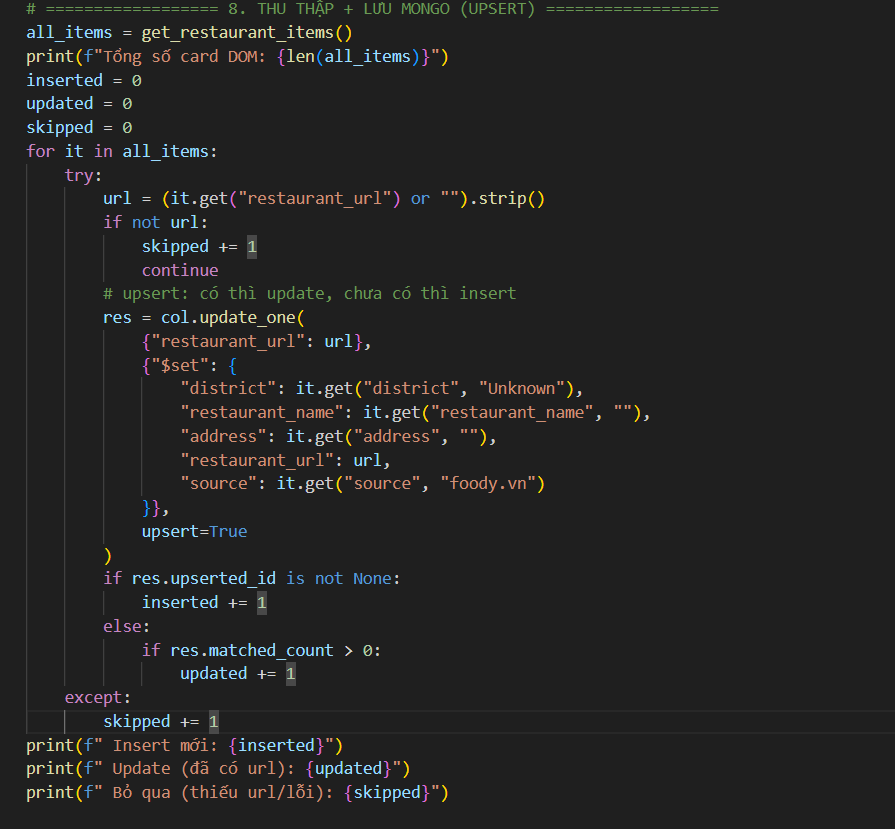
* Dùng selector div.content-item để lấy danh sách “card” quán.
* Với mỗi card:
  + Lấy restaurant\_url từ thẻ <a> (ưu tiên div.title a, fallback a.ng-binding nếu cần).
  + Lấy restaurant\_name từ text tiêu đề.
  + Lấy address từ div.desc.
* Mục tiêu là tạo bản ghi chuẩn gồm: restaurant\_url, restaurant\_name, address.



Hình 3.2.6 : Hàm tách quận/huyện từ địa chỉ quán

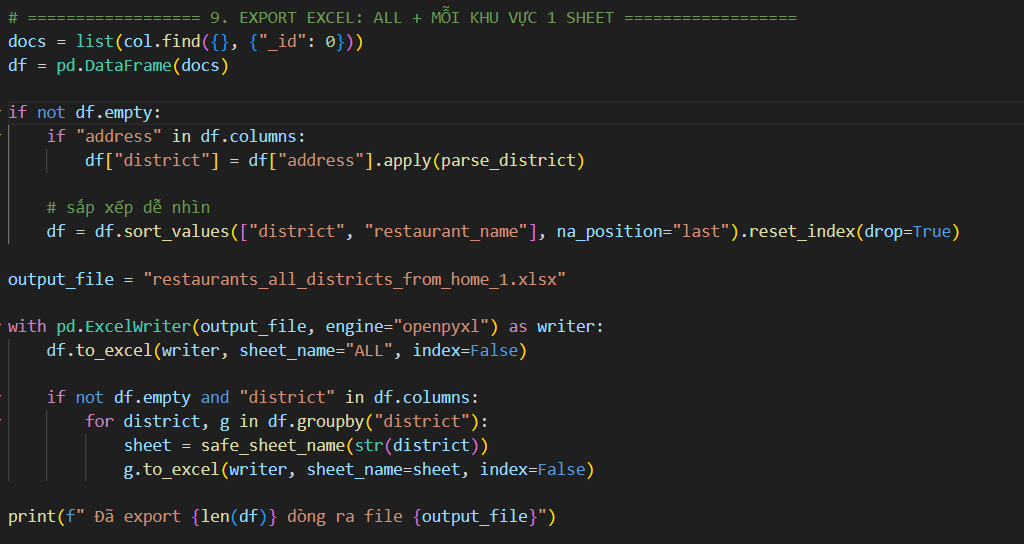
* Chuẩn hóa địa chỉ (loại ký tự thừa, đồng nhất chữ hoa/thường nếu cần).
* Tách được các dạng phổ biến:
  + “Quận …”, “Huyện …”, “TP. Thủ Đức”
  + dạng rút gọn “Q.1”, “Q1”, …
* Nếu không tách được → gán “Unknown” (để xử lý/giảm dần ở bước export).

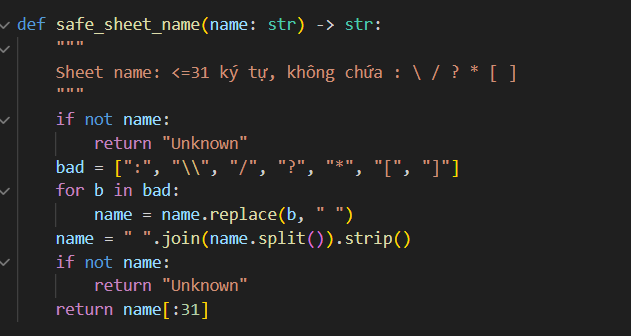




Hình 3.2.7 : Lưu dữ liệu MongoDB theo cơ chế UPSERT

* Tạo unique index theo restaurant\_url để chống trùng dữ liệu.
* Với mỗi quán:
  + Nếu thiếu restaurant\_url → bỏ qua (tránh record rỗng).
  + Nếu có → update\_one(..., upsert=True):
    - Chưa tồn tại → insert mới
    - Đã tồn tại → update thông tin

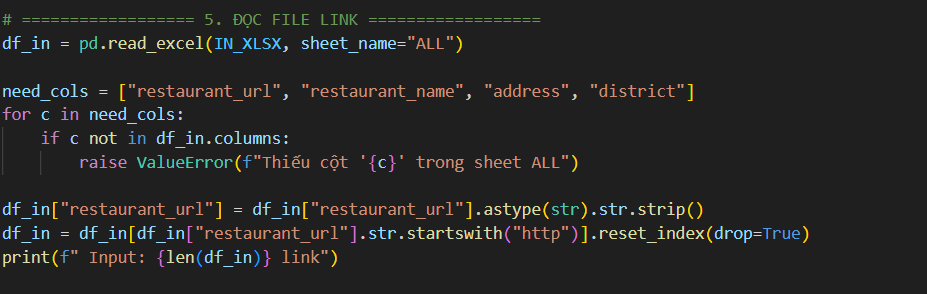




Hình 3.2.8: Xuất file Excel theo quận/huyện

* Đọc toàn bộ dữ liệu từ MongoDB đưa về DataFrame.
* Tính lại district từ address để giảm “Unknown” cho dữ liệu cũ.
* Sắp xếp theo district, restaurant\_name để dễ đọc.
* Xuất file Excel gồm:
  + Sheet ALL tổng hợp
  + Mỗi quận/huyện 1 sheet riêng
* Dùng safe\_sheet\_name() để tránh lỗi tên sheet (ký tự cấm, vượt quá 31 ký tự). Hàm này làm sạch tên quận/huyện trước khi ghi nhiều sheet.

#### *3.2.3.2 File 2: review\_quan\_restaurants\_all.py – Cào bảng review\_quán*

**

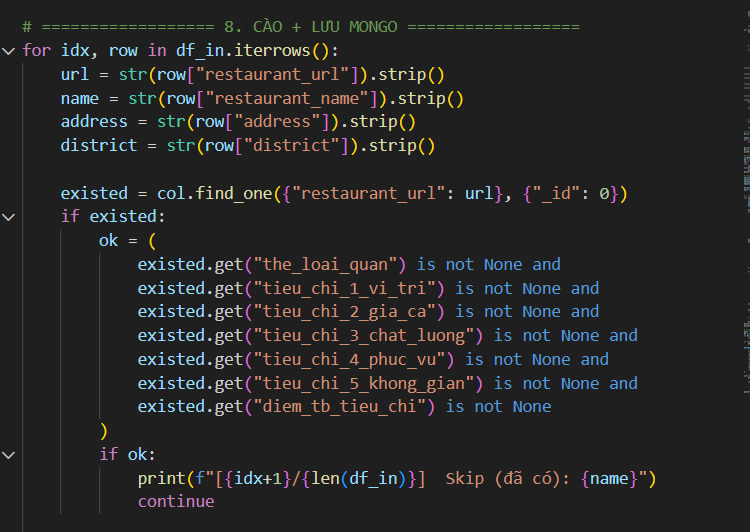
Hình 3.2.9: Đọc file Excel danh sách quán và kiểm tra dữ liệu đầu vào

* Đọc restaurants\_all\_districts\_from\_home\_1.xlsx (sheet ALL).
* Kiểm tra các cột bắt buộc: restaurant\_url, restaurant\_name, address, district.
* Nếu thiếu → dừng chương trình và báo lỗi để tránh crawl sai dữ liệu.

**

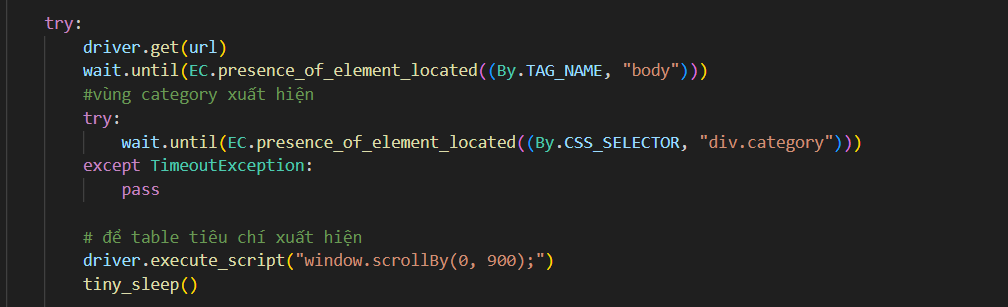
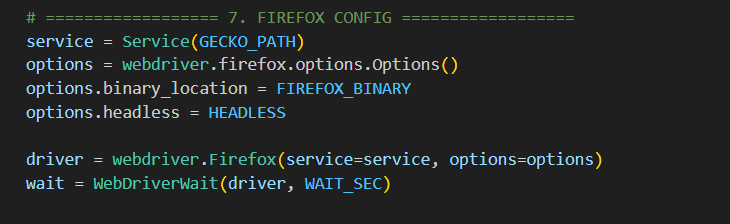
Hình 3.2.10:Khởi tạo MongoDB cho bảng review\_quán và tạo unique index chống trùng

* Dùng MongoDB làm nơi lưu kết quả cào (giảm trùng và dễ truy vấn).
* Tạo unique index theo restaurant\_url giúp mỗi quán chỉ có 1 record review\_quán.

**

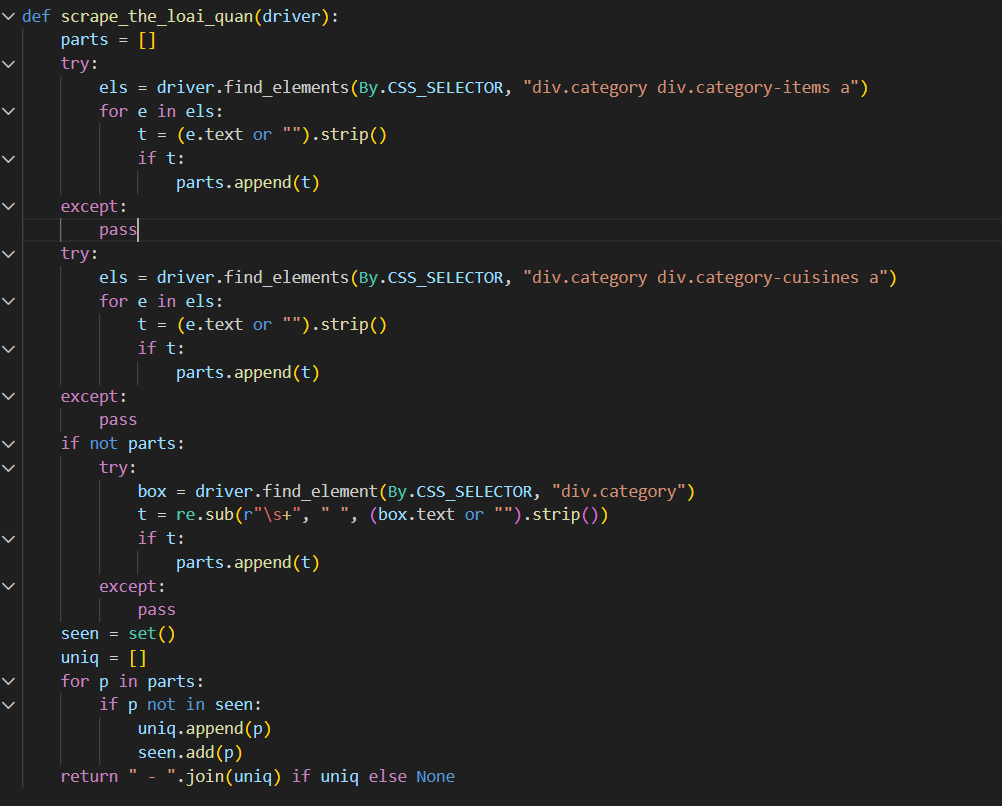
Hình 3.2.11: Cơ chế skip quán đã có đủ dữ liệu (tối ưu thời gian crawl)

* Trước khi mở trang quán, chương trình query Mongo để xem quán đã có:
  + the\_loai\_quan
  + 5 tiêu chí: vị trí/giá cả/chất lượng/phục vụ/không gian
  + diem\_tb\_tieu\_chi
* Nếu đủ → skip, tránh crawl lại gây tốn thời gian.

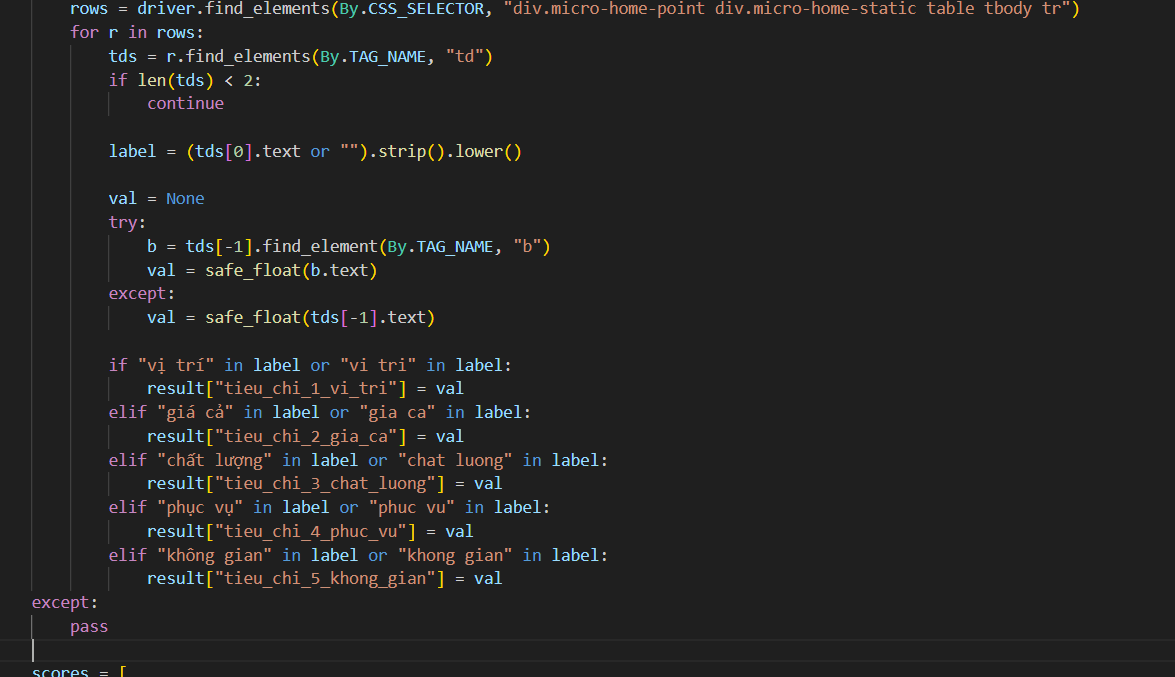
**

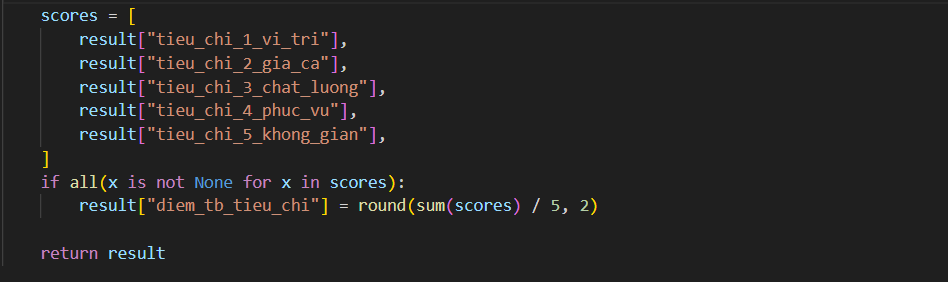
Hình 3.2.12: Mở trang quán, chờ tải và cuộn để hiện bảng tiêu chí

* driver.get(url) để truy cập trang chi tiết quán.
* Chờ body hoặc vùng div.category xuất hiện (timeout thì vẫn tiếp tục để tránh dừng toàn bộ).
* Cuộn trang (scrollBy) để phần điểm/tiêu chí hiển thị đầy đủ

Hình 3.2.13: Hàm lấy thể loại quán

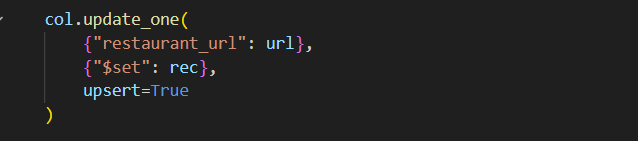
* Thu thập thể loại từ nhiều selector khác nhau để tăng độ ổn định.
* Loại trùng và nối bằng " - " để tạo 1 chuỗi thể loại đầy đủ cho quán.

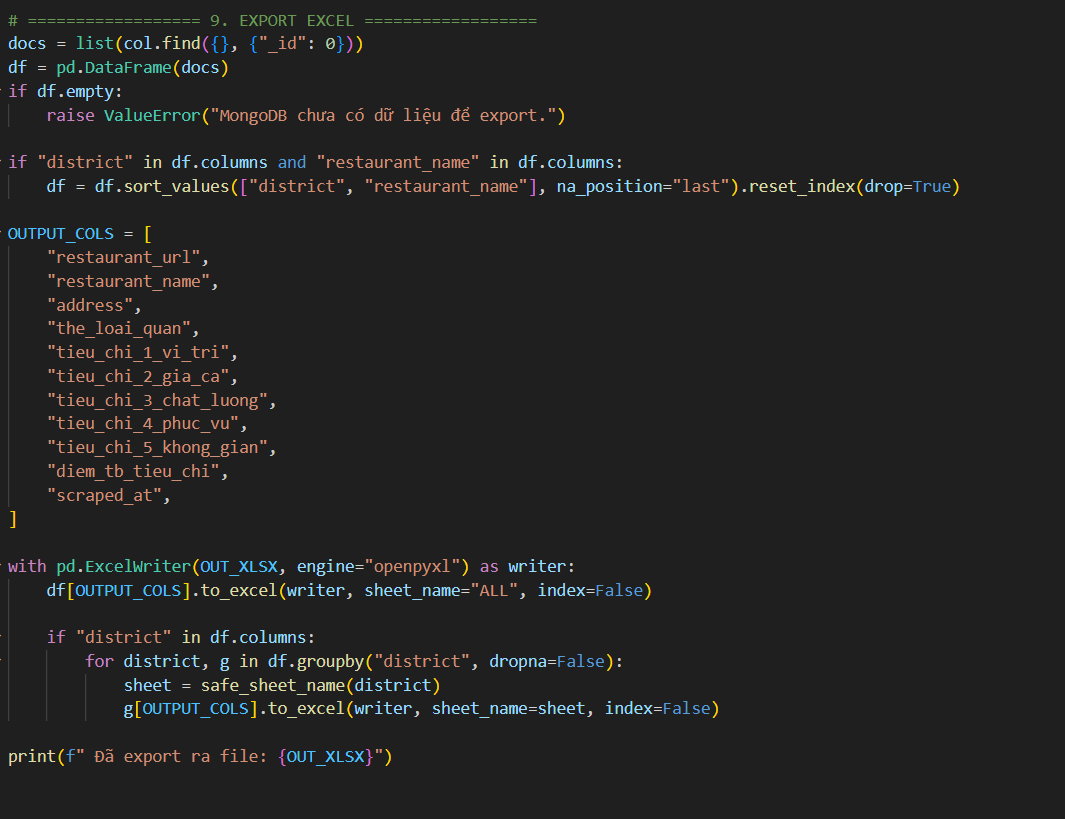
**

**

Hình 3.2.14 : Hàm đọc bảng 5 tiêu chí và tính điểm trung bình

* Duyệt các dòng tr trong bảng điểm.
* Map label về 5 cột chuẩn:
  + tieu\_chi\_1\_vi\_tri, tieu\_chi\_2\_gia\_ca, tieu\_chi\_3\_chat\_luong, tieu\_chi\_4\_phuc\_vu, tieu\_chi\_5\_khong\_gian
* Dùng hàm safe\_float() chuyển điểm về số (xử lý dấu phẩy/dấu chấm).
* Nếu đủ 5 tiêu chí → tính diem\_tb\_tieu\_chi = round(avg, 2).

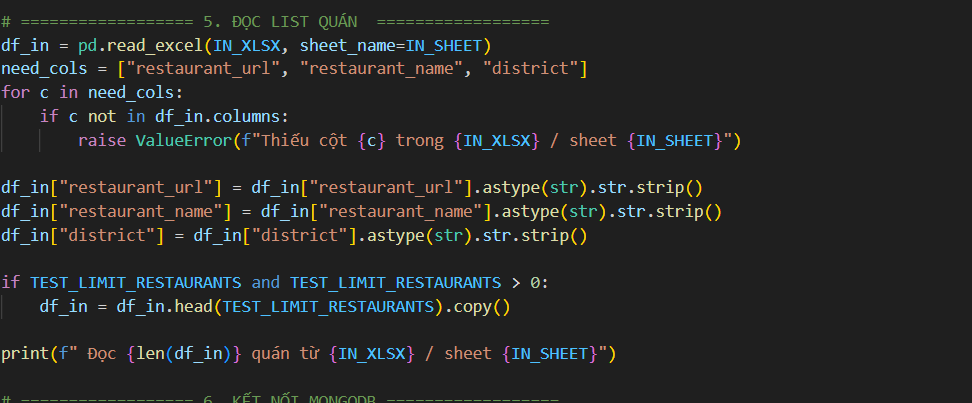
**

**

Hình 3.2.15: Lưu kết quả và xuất Excel

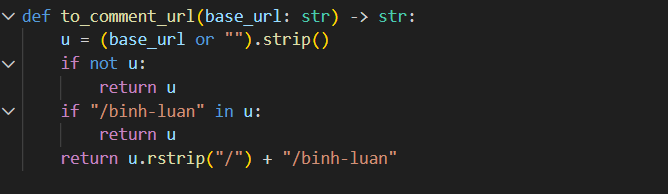
* Mỗi record lưu kèm thông tin quán + kết quả crawl + scraped\_at.
* Sau khi crawl xong:
  + Nếu Mongo rỗng → báo lỗi “chưa có dữ liệu để export”.
  + Nếu có → sort theo district, restaurant\_name và xuất:
    - Sheet ALL
    - Sheet theo từng quận/huyện

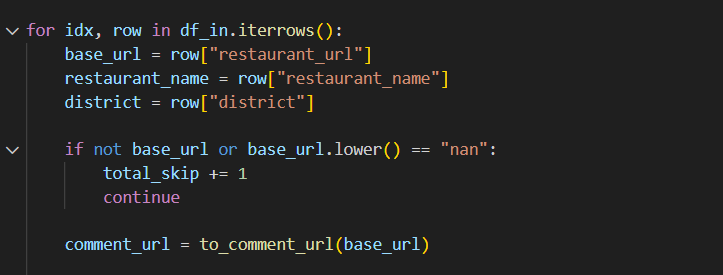
#### *3.2.3.3 File 3: review\_user\_all.py – Cào bảng review\_user*

**

Hình 3.2.16 : Chuẩn bị danh sách quán trước khi cào review\_user

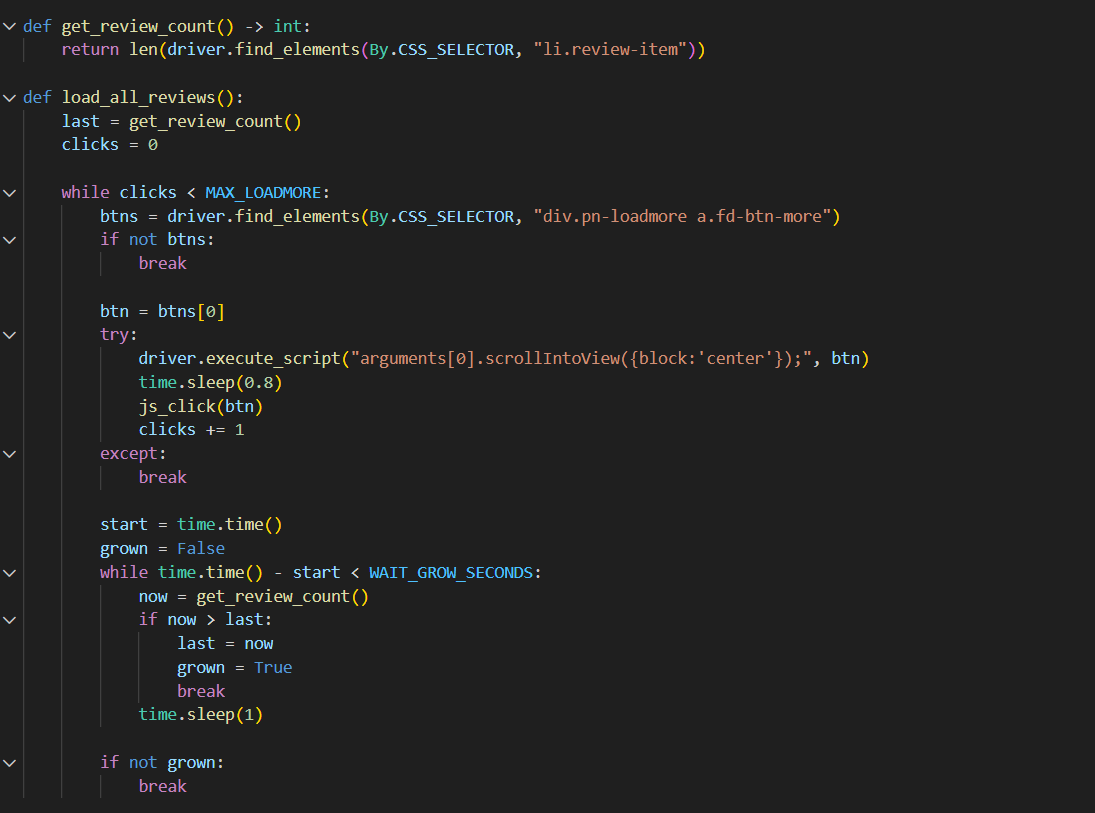
* Đọc file danh sách quán (sheet ALL).
* Kiểm tra các cột bắt buộc: restaurant\_url, restaurant\_name, district.
* Làm sạch chuỗi (xóa khoảng trắng thừa) để hạn chế lỗi link.
* Nếu cần test nhanh → giới hạn số quán đầu tiên bằng TEST\_LIMIT\_RESTAURANTS.

**

**

Hình 3.2.17: Tạo đường dẫn trang bình luận của quán

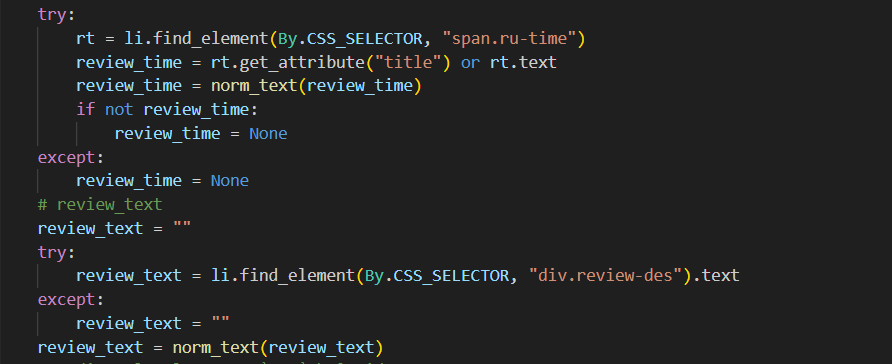
* Một số URL là trang chủ quán, chương trình tự nối thêm /binh-luan.
* Đảm bảo truy cập đúng khu vực chứa danh sách review.



Hình 3.2.18 :Cơ chế bấm “Xem thêm bình luận” và kiểm tra tăng dữ liệu

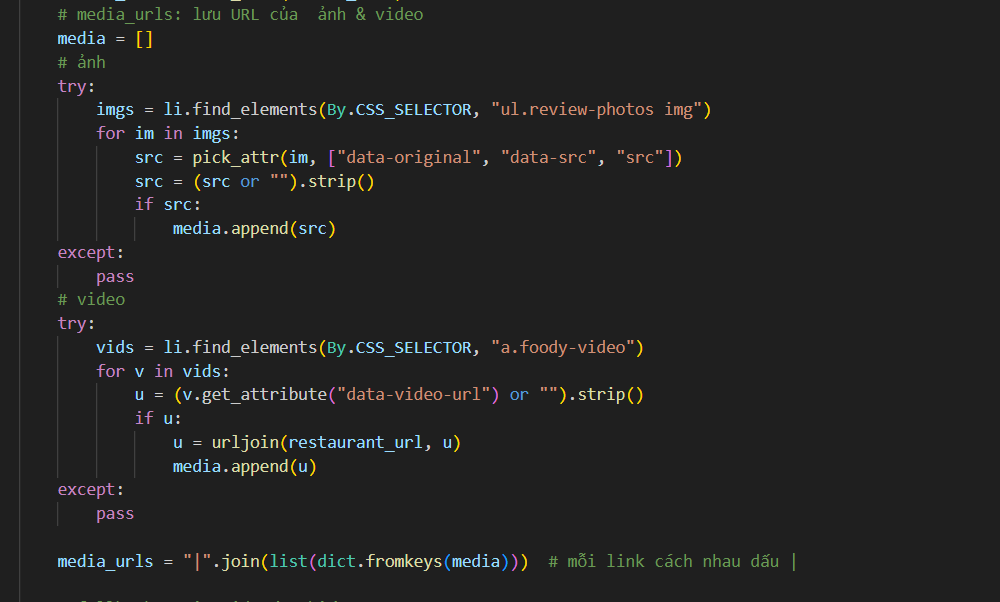
* Vì bình luận được tải động, chương trình phải click load-more nhiều lần.
* Sau mỗi lần click:
  + Chờ số lượng li.review-item tăng.
  + Nếu không tăng sau WAIT\_GROW\_SECONDS hoặc vượt MAX\_LOADMORE → dừng.

**

**

Hình 3.2.19:Trích xuất dữ liệu chính của một bình luận

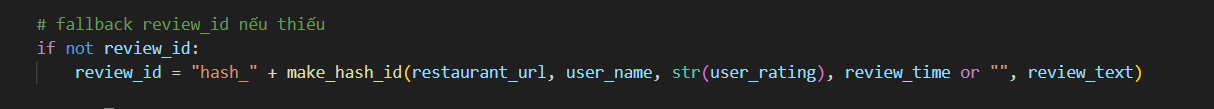
* Lấy user\_name từ thẻ username.
* Lấy user\_rating và chuẩn hoá định dạng số (đổi “,” → “.” nếu cần).
* Lấy review\_time (ưu tiên attr title, fallback text).
* Lấy review\_text từ vùng nội dung bình luận.

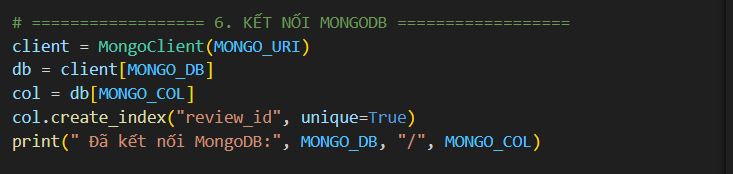
**

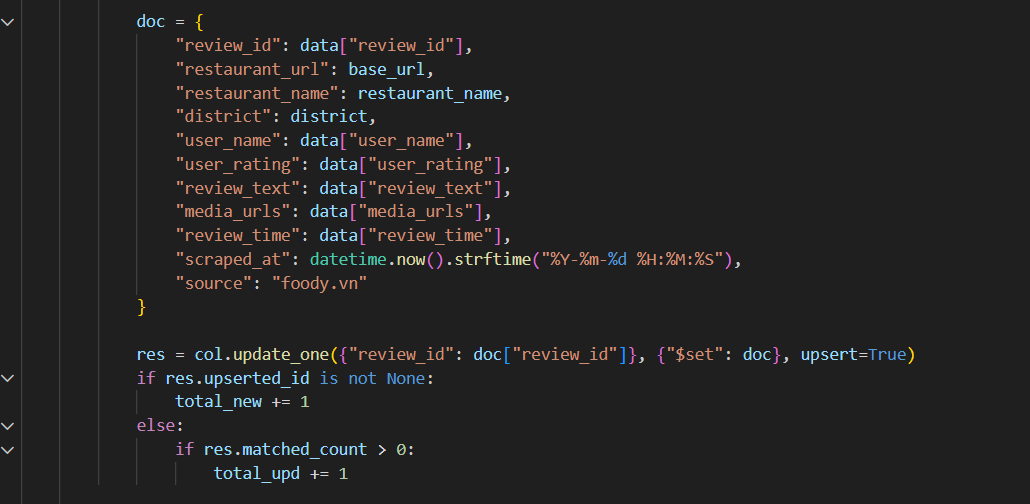
Hình 3.2.20 :Thu thập link ảnh/video trong bình luận

* Ảnh có thể nằm ở nhiều thuộc tính (data-original, data-src, src) nên chương trình thử lần lượt.
* Video lấy từ data-video-url hoặc link liên quan trong thẻ video.
* Dùng urljoin để ghép link tương đối thành link đầy đủ.
* Gộp tất cả vào media\_urls, nối bằng dấu | và loại trùng.



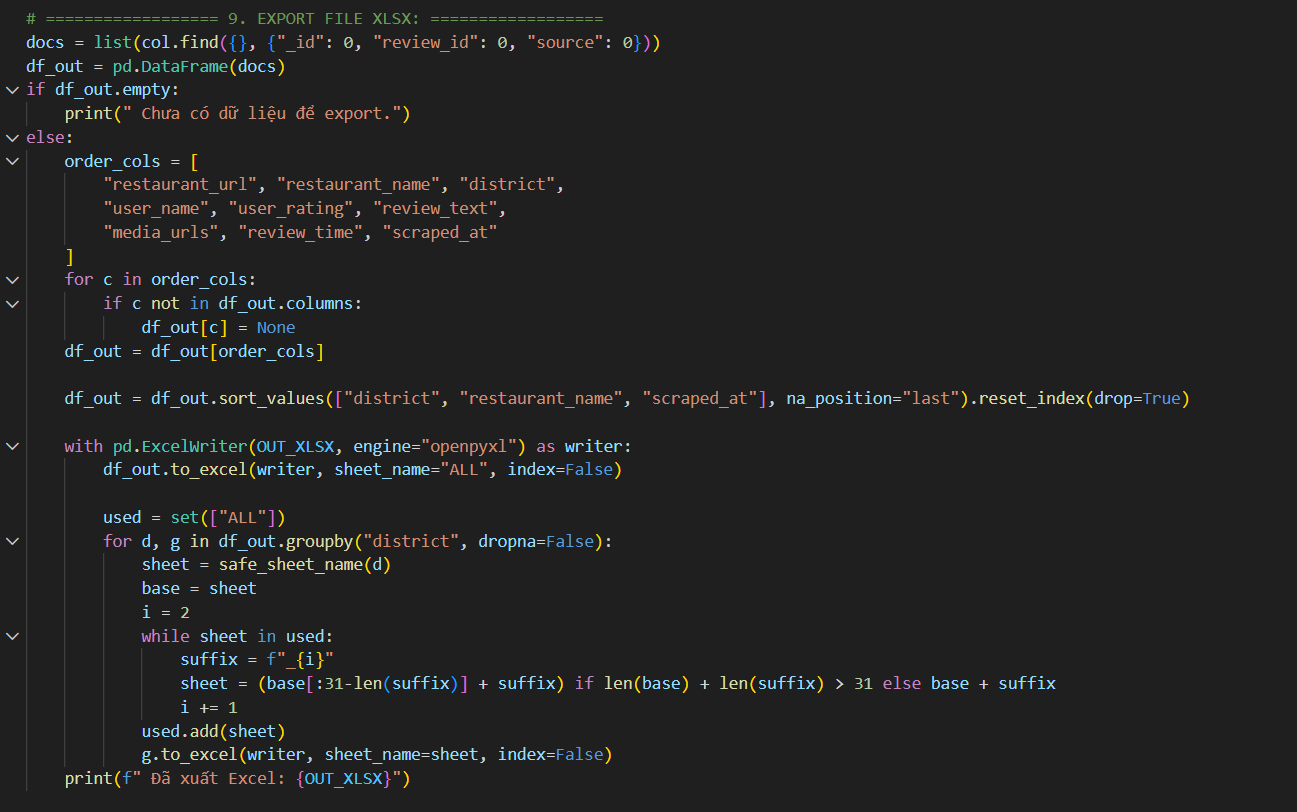
**

**



Hình 3.2.21 :Tạo khóa duy nhất review\_id và lưu MongoDB

* Nếu trang có data-review → dùng trực tiếp làm review\_id.
* Nếu không có → tạo hash để đảm bảo mỗi bình luận có ID duy nhất.
* Tạo unique index theo review\_id nhằm chống trùng review.
* Lưu record theo upsert, đồng thời đếm total\_new, total\_upd, total\_err để thống kê tiến độ.



Hình 3.2.22:Xuất file review\_user\_all.xlsx theo district

* Loại một số cột nội bộ như \_id, review\_id, source khi xuất file.
* Sắp xếp theo district, restaurant\_name, scraped\_at để dễ xem.
* Xuất:
  + Sheet ALL
  + Sheet theo từng quận/huyện
* Nếu trùng tên sheet (do giới hạn 31 ký tự hoặc tên giống nhau) → thêm hậu tố \_2,\_3 để tránh lỗi.

## **3.3 Mô tả kết quả thu thập dữ liệu và cấu trúc dữ liệu**

Sau quy trình thực hiện, dữ liệu thu thập được tổ chức thành 03 bảng chính, tương ứng với 03 giai đoạn crawl

### *3.3.1 Bảng 1 – Danh sách quán ăn*

* Nguồn tạo ra: File 1 – restaurants\_all\_districts\_from\_home\_1.py
* Mục đích: Thu thập danh sách quán làm nền tảng để crawl sâu ở các bước tiếp theo.
* Input
  + Trang web Foody (sau khi đăng nhập), danh sách quán hiển thị theo dạng feed.
  + HTML động (cần thao tác load-more).
* Output
  + MongoDB: foody\_db.restaurants\_all (lưu dạng document, chống trùng theo restaurant\_url).
  + Excel: restaurants\_all\_districts\_from\_home\_1.xlsx
* Sheet ALL và sheet theo từng district (quận/huyện)
* Cấu trúc dữ liệu (các cột chính)

|  |  |  |
| --- | --- | --- |
| Tên biến | Mô tả | Kiểu dữ liệu |
| restaurant\_url | Link quán (khóa chính/unique để upsert) | string |
| restaurant\_name | Tên quán | string |
| address | Địa chỉ quán | string |
| district | Quận/huyện (được tách từ address bằng hàm parse\_district) | string |
| scraped\_at | Thời điểm thu thập dữ liệu | datetime |

* Ý nghĩa: Bảng 1 đóng vai trò “bảng gốc”, giúp nhóm có được danh sách link quán đầy đủ để tiếp tục thu thập dữ liệu đánh giá ở bảng 2 và bảng 3.

### *3.3.2 Bảng 2 – review\_quán*

* Nguồn tạo ra: File 2 – review\_quan\_restaurants\_all.py
* Mục đích: Thu thập thông tin đánh giá tổng quan cho từng quán, phục vụ so sánh mức độ đánh giá giữa các quán hoặc giữa các quận.
* Input
  + Excel đầu vào: restaurants\_all\_districts\_from\_home\_1.xlsx (sheet ALL)
  + Các cột bắt buộc: restaurant\_url, restaurant\_name, address, district
  + Trang chi tiết từng quán trên Foody (không cần đăng nhập)
* Output
  + MongoDB: review\_quan\_db.review\_restaurants\_all
    - Unique theo restaurant\_url (mỗi quán 1 record review\_quán)
  + Excel: review\_quan\_restaurants\_all.xlsx
    - Sheet ALL và sheet theo từng district (quận/huyện)
* Cấu trúc dữ liệu (các cột chính)

|  |  |  |
| --- | --- | --- |
| Tên biến | Mô tả | Kiểu dữ liệu |
| restaurant\_url | Link quán (khóa chính/unique để upsert) | string |
| restaurant\_name | Tên quán | string |
| address | Địa chỉ quán | string |
| district | Quận/huyện (được tách từ address bằng hàm parse\_district) | string |
| the\_loai\_quan | Thể loại quán (gộp nhiều tag, nối bằng “ - ”) | string |
| tieu\_chi\_1\_vi\_tri | Điểm tiêu chí “Vị trí” | float |
| tieu\_chi\_2\_gia\_ca | Điểm tiêu chí “Giá cả” | float |
| tieu chi 3\_chat\_luong | Điểm tiêu chí “Chất lượng” | float |
| tieu\_chi\_4\_phuc\_vu | Điểm tiêu chí “Phục vụ” | float |
| tieu\_chi\_5\_khong\_gian | Điểm tiêu chí “Không gian” | float |
| diem\_tb\_tieu\_chi | Điểm trung bình 5 tiêu chí (chỉ tính khi đủ dữ liệu) | float |
| scraped\_at | Thời điểm thu thập dữ liệu | datetime |

* Ý nghĩa: Bảng 2 phản ánh “chất lượng tổng quan” của quán dựa trên 5 khía cạnh quan trọng, giúp nhóm dễ làm EDA theo hướng: quận nào có điểm cao, tiêu chí nào thường thấp/cao,…

### *3.3.3 Bảng 3 – review\_user*

* Nguồn tạo ra: File 3 – review\_user\_all.py
* Mục đích: Thu thập bình luận người dùng và nội dung review để phân tích sâu (tần suất review, phân bố điểm, sentiment đơn giản, ảnh/video,…).
* Input
  + Excel đầu vào: restaurants\_all\_districts\_from\_home\_1.xlsx (sheet ALL)
  + Các cột bắt buộc: restaurant\_url, restaurant\_name, district
  + Trang bình luận từng quán: URL /binh-luan (có load-more động)
* Output
  + MongoDB: review\_quan\_db.review\_user\_all
    - Unique theo review\_id (mỗi bình luận 1 record)
  + Excel: review\_user\_all.xlsx
    - Sheet ALL và sheet theo từng district (quận/huyện)
* Cấu trúc dữ liệu (các cột chính)

|  |  |  |
| --- | --- | --- |
| Tên biến | Mô tả | Kiểu dữ liệu |
| restaurant\_url | Link quán (khóa chính/unique để upsert) | string |
| restaurant\_name | Tên quán | string |
| district | Quận/huyện (được tách từ address bằng hàm parse\_district) | string |
| review\_id | Mã định danh review (lấy từ data-review; nếu thiếu tạo hash md5 dự phòng) | string |
| user\_name | Tên người dùng đăng bình luận | string |
| user\_rating | Điểm người dùng đánh giá | float |
| review\_text | Nội dung bình luận | string |
| review\_time | Thời gian đăng bình luận (lấy từ title/text) | string |
| media\_urls | Link ảnh/video trong review (gộp, nối bằng “|”, loại trùng) | string |
| scraped\_at | Thời điểm thu thập dữ liệu | datetime |

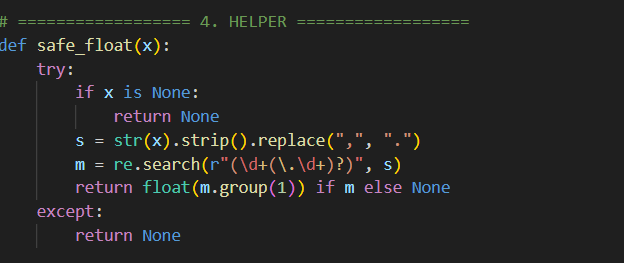
* Ý nghĩa: Bảng 3 là bảng “dữ liệu thô chi tiết” theo từng bình luận, dùng tốt cho EDA theo hướng hành vi người dùng, phân bố rating, phân tích text (tích cực/tiêu cực), quán nào nhiều review nhất,…

## **3.4 Phương pháp tiền xử lý dữ liệu**



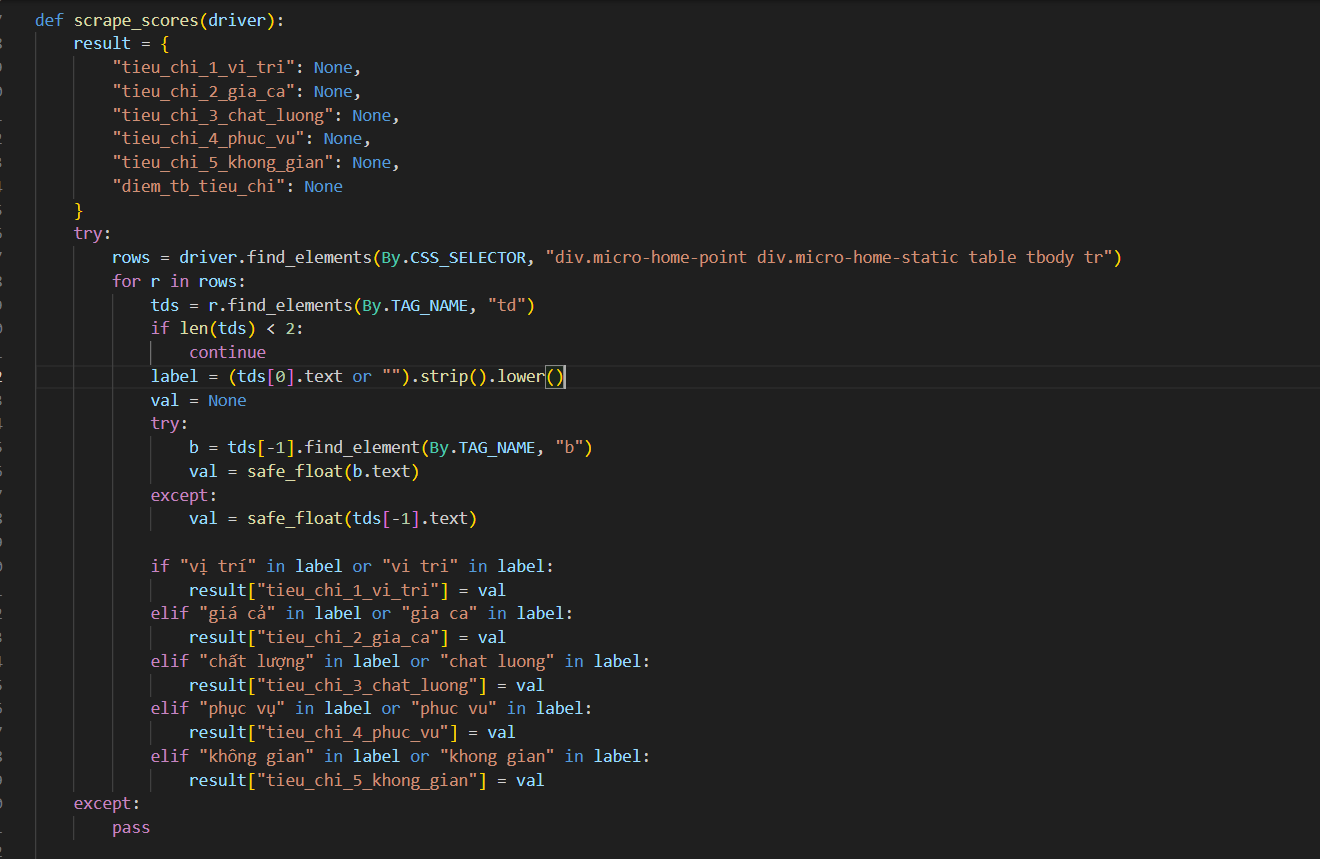
Hình 3.4.1 : Tiền xử lý chuỗi địa chỉ trước khi trích xuất quận/huyện

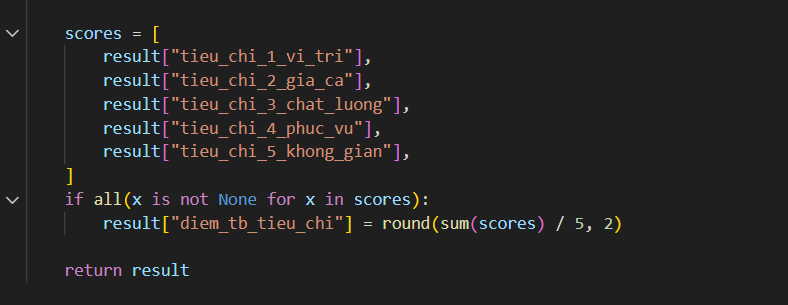
* Xóa khoảng trắng thừa và chuẩn hóa chuỗi địa chỉ trước khi xử lý.
* Tách địa chỉ theo dấu phẩy, ưu tiên phần bắt đầu bằng “Quận/Huyện”.
* Nhận diện trường hợp đặc biệt “Thủ Đức” và xử lý các dạng viết tắt Q.1/Q1.
* Nếu không xác định được → trả về “Unknown” để tránh sai dữ liệu.



Hình 3.4.2: Chuẩn hóa dữ liệu điểm số

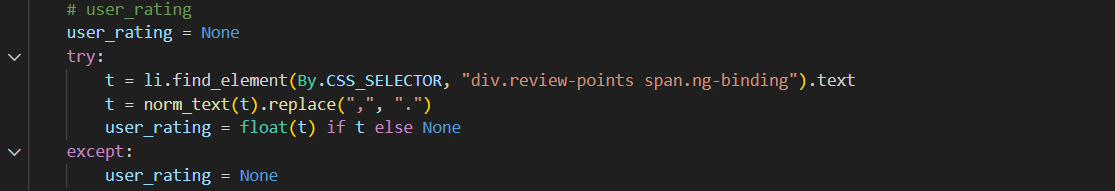
* Xử lý trường hợp điểm có dấu “,” (ví dụ 8,5) bằng cách đổi sang “.”
* Dùng regex để trích số trong chuỗi (tránh dính chữ/ký tự khác).
* Nếu không trích được số hoặc lỗi → trả về None để tránh crash chương trình.





Hình 3.4.3 : Quy trình chuẩn hóa và tổng hợp điểm 5 tiêu chí đánh giá

* Làm sạch tên tiêu chí bằng strip().lower() trước khi so khớp nhãn.
* Trích điểm từ HTML và chuyển về số thực bằng safe\_float() để hạn chế lỗi định dạng.
* Ánh xạ điểm vào 5 tiêu chí: vị trí, giá cả, chất lượng, phục vụ, không gian.
* Chỉ tính diem\_tb\_tieu\_chi khi đủ 5 tiêu chí hợp lệ (khác None).
* Tính trung bình và làm tròn 2 chữ số để tạo chỉ số tổng quan cho quán..



Hình 3.4.4 : Chuẩn hóa user\_rating trong bảng review\_user

* Lấy text điểm đánh giá từ DOM bằng CSS selector div.review-points span.ng-binding.
* Chuẩn hoá dấu thập phân: thay “,” thành “.” để phù hợp khi ép kiểu số thực.
* Ép kiểu float(t) để đưa user\_rating về dạng số, phục vụ thống kê và vẽ biểu đồ.
* Bọc try/except: nếu lỗi hoặc không đọc được → gán None để chương trình vẫn chạy tiếp..

## **3.5 Phương pháp phân tích dữ liệu**

### *3.5.1 Kết quả thu thập dữ liệu*

Dữ liệu được thu thập từ website Foody.vn và được tổ chức thành 03 bảng chính (tương ứng 03 giai đoạn crawl). Kết quả thu thập đạt được tổng cộng 575 quán ăn và 19.852 bình luận của người dùng.

Dữ liệu được lưu trữ trong MongoDB để phục vụ cho các bước phân tích tiếp theo, cụ thể:

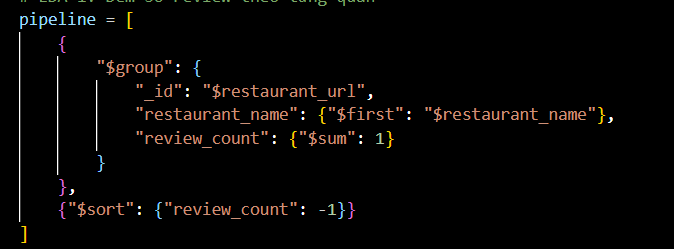
* Bảng 1 – Danh sách quán (restaurants\_all): 575 documents
* Bảng 2 – Review\_quán (review\_restaurants\_all): 575 documents
* Bảng 3 – Review\_user (review\_user\_all): 19.852 documents

Các thuộc tính thu thập được bao gồm:

* Bảng 1 – Danh sách quán ăn
  + restaurant\_url: Đường dẫn quán (unique)
  + restaurant\_name: Tên quán
  + address: Địa chỉ quán
  + district: Quận/Huyện (tách từ địa chỉ)
  + source: nguồn dữ liệu (Foody.vn)
* Bảng 2 – Review\_quán
  + restaurant\_url: Đường dẫn quán (unique)
  + restaurant\_name: Tên quán
  + address: Địa chỉ quán
  + district: Quận/Huyện
  + the\_loai\_quan: Thể loại quán
  + tieu\_chi\_1\_vi\_tri: Điểm vị trí
  + tieu\_chi\_2\_gia\_ca: Điểm giá cả
  + tieu\_chi\_3\_chat\_luong: Điểm chất lượng
  + tieu\_chi\_4\_phuc\_vu: Điểm phục vụ
  + tieu\_chi\_5\_khong\_gian: Điểm không gian
  + diem\_tb\_tieu\_chi: Điểm trung bình 5 tiêu chí
  + scraped\_at: Thời điểm crawl
* Bảng 3 – Review\_user
  + review\_id: Mã review (unique)
  + restaurant\_url: Đường dẫn quán
  + restaurant\_name: Tên quán
  + district: Quận/Huyện
  + user\_name: Tên người dùng
  + user\_rating: Điểm đánh giá của người dùng
  + review\_text: Nội dung bình luận
  + review\_time: Thời gian đăng bình luận
  + media\_urls: Link ảnh/video (nếu có)
  + scraped\_at: Thời điểm crawl
  + source: nguồn dữ liệu (Foody.vn)

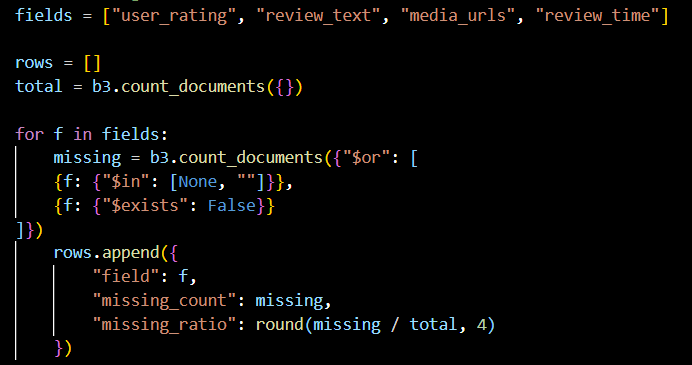
### *3.5.2 Phương pháp phân tích*

Sau khi hoàn tất quá trình thu thập và lưu trữ dữ liệu vào hệ thống MongoDB, nhóm chúng tôi tiếp tục thực hiện bước phân tích dữ liệu khám phá (EDA – Exploratory Data Analysis) nhằm đánh giá tổng quan chất lượng dữ liệu cũng như nhận diện các xu hướng nổi bật. Mục tiêu của giai đoạn này là giúp hiểu rõ đặc điểm hành vi của người dùng và mức độ phổ biến của các quán ăn trên Foody.vn thông qua các yếu tố như: số lượng đánh giá, điểm số trung bình, mức độ tương tác của người dùng và phân bố dữ liệu theo từng quán.

* **Có bao nhiêu quán được review và mỗi quán trung bình có bao nhiêu review?**

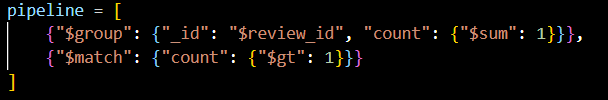
Hình 3.5.1: Thống kê số lượng đánh giá theo từng quán

Sau khi thực hiện truy vấn trên, chúng tôi thu được danh sách các quán ăn cùng tổng số lượt đánh giá tương ứng. Điều này giúp nhóm nghiên cứu hiểu rõ mức độ quan tâm của người dùng đối với từng quán, từ đó có thể nhận diện những quán có độ phổ biến cao cũng như những quán chưa có nhiều đánh giá để cân nhắc trong phân tích tiếp theo.

* **Có dữ liệu bị thiếu (missing values) không? Thiếu ở cột nào nhiều nhất?**

Hình 3.5.2: Truy vấn kiểm tra tỷ lệ dữ liệu thiếu trong các trường quan trọng

Nội dung truy vấn trên được sử dụng để phát hiện những trường thông tin không được người dùng cung cấp đầy đủ. Qua đó, nhóm có thể đánh giá mức độ hoàn thiện của dữ liệu, từ đó đưa ra các biện pháp xử lý phù hợp như loại bỏ, bổ sung hoặc thay thế dữ liệu thiếu nhằm đảm bảo chất lượng phân tích.

* **Kiểm tra dữ liệu trùng lặp: có review nào bị trùng không?**

Hình 3.5.3: Kiểm tra trùng lặp

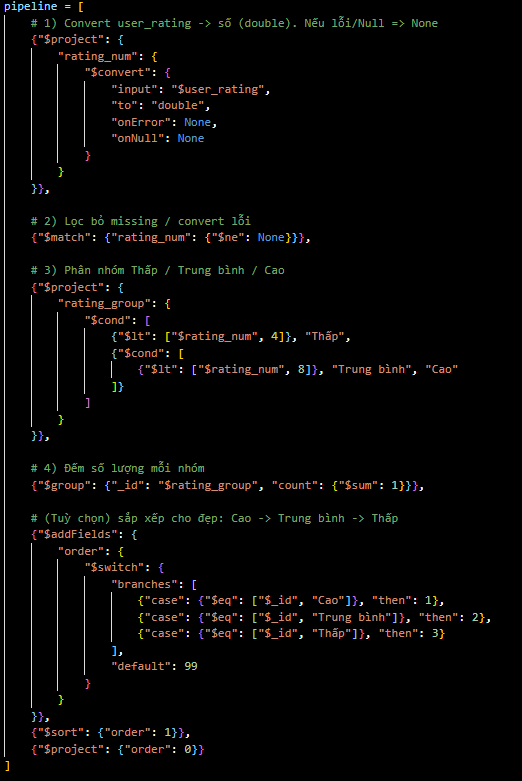
Truy vấn này giúp xác định các đánh giá bị ghi trùng trong quá trình thu thập dữ liệu. Việc kiểm tra sự trùng lặp là rất quan trọng để đảm bảo tính chính xác của các phân tích sau này, cũng như tránh làm sai lệch kết quả đánh giá từ phía người dùng.

* **Phân phối điểm user\_rating (1–10) như thế nào?**

****

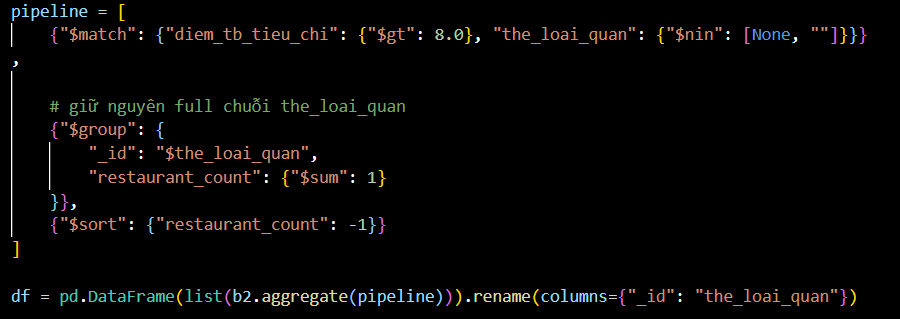
Hình 3.5.4: Phân tích sự phân bố điểm đánh giá của người dùng

Nhờ truy vấn này, chúng tôi có thể quan sát xu hướng chấm điểm của người dùng. Kết quả phân tích giúp nhận diện nhóm đánh giá tích cực hoặc tiêu cực, hỗ trợ tốt hơn trong việc đưa ra chiến lược đề xuất hợp lý.

* **Rating có bị lệch về phía tích cực hay tiêu cực không?**

Hình 3.5.5: Phân loại đánh giá theo mức độ hài lòng của người dùng

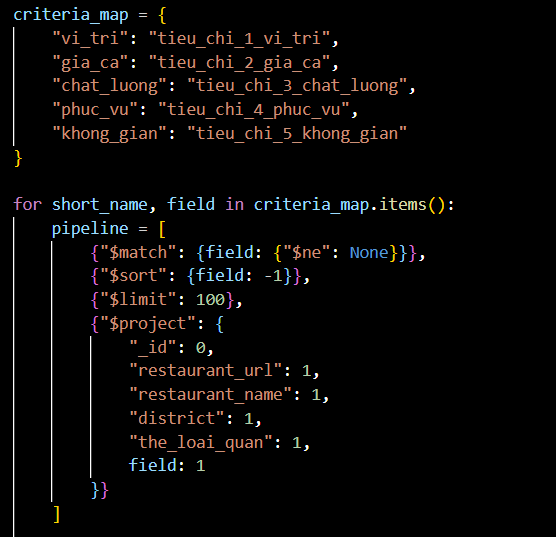
Chúng tôi chia các đánh giá thành ba nhóm: thấp, trung bình và cao. Điều này giúp tập trung phân tích từng phân khúc người dùng, từ đó đưa ra các nhận định phù hợp cho từng mức độ hài lòng.

* **Thể loại quán nào CÓ diem\_tb\_tieu\_chi TRÊN 8.0?**

Hình 3.5.6: Tìm ra thể loại quá nào có điểm trung bình tiêu chí trên 8.0

Qua truy vấn này, nhóm nghiên cứu có thể xác định được những thể loại quán được người dùng quan tâm nhiều hơn. Điều này giúp ích cho việc đề xuất các nhà hàng phù hợp với xu hướng thị trường và sở thích của khách hàng.

* **Quán nào nổi bật theo từng tiêu chí (Vị trí / Giá cả / Chất lượng / Phục vụ / Không gian)?**

****

Hình 3.5.7: Top 100 quán theo từng tiêu chí

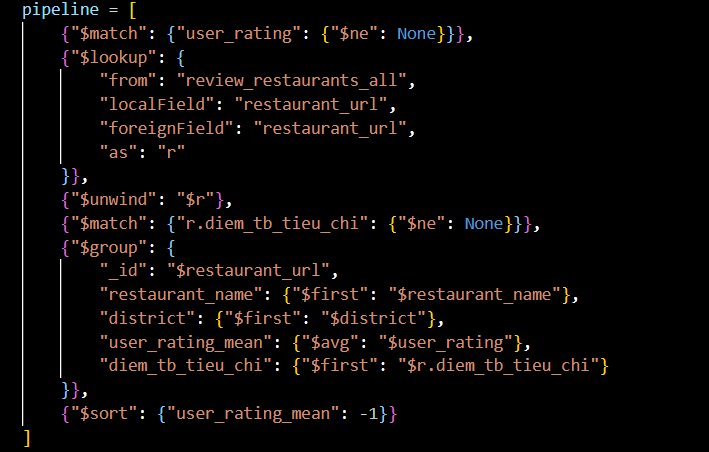
Truy vấn này giúp làm rõ những khía cạnh nổi bật của từng nhà hàng dựa trên phản hồi của khách hàng. Từ đó đưa ra các đề xuất phù hợp theo từng nhu cầu cụ thể, ví dụ như ưu tiên vị trí, không gian hay chất lượng món ăn.

Quán nào có điểm cao nhưng số lượng review thấp (có nguy cơ “ảo điểm”)?

****

Hình 3.5.8: Tìm ra quán có điểm cao nhưng review thấp

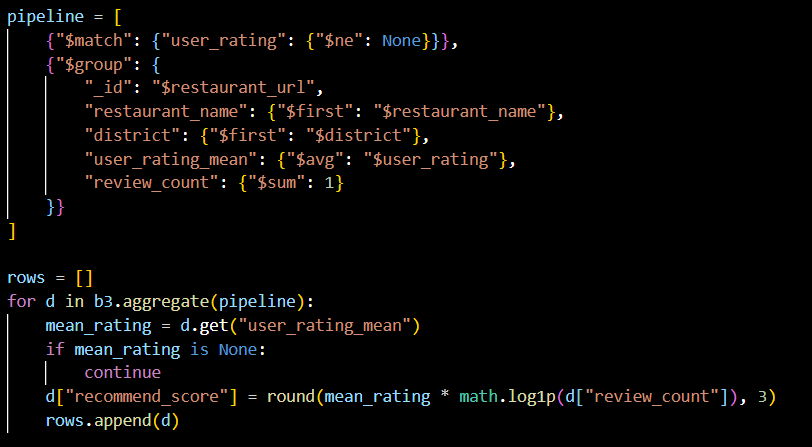
Nhóm tiến hành nhận diện các quán có điểm số nổi bật nhưng lượng đánh giá thấp. Đây có thể là những quán tiềm năng hoặc cũng có thể mang tính rủi ro khi đề xuất do chưa có đủ cơ sở đánh giá từ người dùng.

* **Điểm tiêu chí của quán (diem\_tb\_tieu\_chi) có liên quan đến rating người dùng không?  
  **

Hình 3.5.9: So sánh điểm đánh giá người dùng và điểm tiêu chí của quán ăn

Truy vấn này hỗ trợ đánh giá mức độ nhất quán giữa cảm nhận của người dùng và chất lượng được hệ thống ghi nhận. Khi hai giá trị lệch nhau, nhóm có thể tìm hiểu nguyên nhân để đưa ra nhận định khách quan hơn trong phân tích.

* **Top quán “đáng tin” để đề xuất dựa trên điểm cao và nhiều review**



Hình 3.5.10: Quán đáng tin để đề xuất

Dựa trên điểm đánh giá trung bình và số lượt đánh giá, nhóm tính toán ra mức độ uy tín và độ phổ biến của từng quán. Kết quả này giúp hỗ trợ người dùng lựa chọn quán ăn phù hợp và đáng tin cậy hơn.

### *3.5.4 Kết luận*

Thông qua các câu hỏi truy vấn, nhóm em đã khai thác thành công các thông tin quan trọng cho quy trình phân tích như tên quán, địa chỉ, đánh giá và nội dung bình luận của người dùng trên Foody.vn. Các câu truy vấn được xây dựng linh hoạt, hỗ trợ lọc và tổng hợp dữ liệu theo nhu cầu, từ đó tạo nền tảng cho việc đánh giá chất lượng quán ăn. Kết quả truy vấn cho thấy hệ thống dữ liệu được quản lý hiệu quả và phục vụ tốt mục tiêu đề xuất quán cho người dùng.

# **CHƯƠNG 4: KẾT LUẬN VÀ KIẾN NGHỊ**

## **4.1 Kết luận**

Trong thời đại công nghệ số phát triển mạnh mẽ, nhu cầu tìm kiếm và lựa chọn quán ăn phù hợp ngày càng trở nên quan trọng đối với người dùng. Đề tài *“Phân tích và đề xuất quán ăn dựa trên dữ liệu đánh giá người dùng từ Foody.vn sử dụng công cụ Selenium và MongoDB”* đã đạt được những kết quả thiết thực và có ý nghĩa.

Trước hết, đề tài đã xây dựng được quy trình thu thập dữ liệu tự động từ Foody.vn bằng cách sử dụng Selenium để mô phỏng hành vi người dùng, xử lý các nội dung tải động và trích xuất dữ liệu đánh giá của người dùng một cách hiệu quả. Các thông tin như tên quán, đánh giá, hình ảnh, nội dung nhận xét và thời gian đăng tải được thu thập đầy đủ, góp phần tạo nên tập dữ liệu đa dạng và chân thực.

Tiếp theo, cơ sở dữ liệu MongoDB đã được lựa chọn và triển khai thành công nhằm lưu trữ và quản lý dữ liệu dạng văn bản phi cấu trúc. Việc sử dụng MongoDB giúp quá trình đọc – ghi dữ liệu nhanh hơn, dễ mở rộng và thuận lợi cho phân tích sau này.

Dựa trên dữ liệu thu thập được, đề tài đã tiến hành phân tích các yếu tố liên quan đến đánh giá của người dùng như mức độ hài lòng, xu hướng nhận xét tích cực/tiêu cực… Qua đó đưa ra gợi ý bước đầu cho việc xây dựng hệ thống đề xuất quán ăn, hỗ trợ người dùng lựa chọn địa điểm phù hợp hơn.

Nhìn chung, đề tài đã chứng minh tính khả thi của việc ứng dụng tự động hóa thu thập dữ liệu và công nghệ lưu trữ NoSQL trong khai thác dữ liệu đánh giá trực tuyến, góp phần khẳng định vai trò của phân tích dữ liệu trong nâng cao trải nghiệm ẩm thực.

## **4.2 Kiến nghị**

Khuyến nghị nâng cao chất lượng và tiêu chuẩn dữ liệu trên các nền tảng đánh giá ẩm thực: Foody.vn và các nền tảng tương tự cần xem xét cải tiến cấu trúc dữ liệu, tổ chức lại thông tin đánh giá theo chuẩn thống nhất và rõ ràng hơn. Việc phát triển thêm các API hoặc bộ tiêu chuẩn dữ liệu mở sẽ tạo điều kiện thuận lợi cho các nhà nghiên cứu và các tổ chức khai thác dữ liệu phục vụ phân tích, nghiên cứu thị trường và dự đoán xu hướng ẩm thực trong tương lai. Điều này không chỉ giúp nâng cao tính minh bạch dữ liệu mà còn hỗ trợ mạnh mẽ quá trình chuyển đổi số trong ngành dịch vụ ăn uống.

Khuyến khích các doanh nghiệp ẩm thực đầu tư vào công nghệ dữ liệu và tiếp nhận phản hồi từ khách hàng: Các quán ăn, nhà hàng nên khai thác hiệu quả nguồn dữ liệu đánh giá từ người dùng trên Foody.vn để cải thiện chất lượng dịch vụ, điều chỉnh chiến lược marketing và nâng cao trải nghiệm khách hàng. Việc theo dõi và phân tích dữ liệu đánh giá thường xuyên sẽ giúp doanh nghiệp nắm bắt đúng nhu cầu của người tiêu dùng, từ đó gia tăng khả năng cạnh tranh trong bối cảnh ngành ẩm thực phát triển nhanh chóng và cạnh tranh cao.

Chủ trương mở rộng nghiên cứu cho các ứng dụng thực tế: Trong tương lai, các nghiên cứu có thể kết hợp thêm các phương pháp phân tích nâng cao như xử lý ngôn ngữ tự nhiên (NLP) để phân tích cảm xúc của người dùng, hay tích hợp với các thuật toán học máy để xây dựng hệ thống gợi ý thông minh và cá nhân hóa hơn. Ngoài ra, việc thiết kế giao diện hỗ trợ người dùng tra cứu và lựa chọn quán ăn nhanh chóng theo nhu cầu sẽ giúp kết quả nghiên cứu có giá trị ứng dụng cao, mang lại lợi ích trực tiếp cho cộng đồng

# **TÀI LIỆU THAM KHẢO**

Real Python, “Python Web Scraping: Practical Introduction.” [Online]. Available: https://realpython.com/python-web-scraping-practical-introduction/.

A. Koster et al., “Robots Exclusion Protocol,” RFC 9309, IETF. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9309.html.

Google Developers, “robots.txt Specifications.” [Online]. Available: https://developers.google.com/crawling/docs/robots-txt/robots-txt-spec.

Requests Documentation, “Quickstart.” [Online]. Available: https://requests.readthedocs.io/en/master/user/quickstart/.

Selenium, “WebDriver Waits.” [Online]. Available: https://www.selenium.dev/documentation/webdriver/waits/.

Selenium Python, “Waits.” [Online]. Available: https://selenium-python.readthedocs.io/waits.html.

MongoDB Manual, “Aggregation Pipeline.” [Online]. Available: https://www.mongodb.com/docs/manual/core/aggregation-pipeline/.

MongoDB Manual, “$lookup (aggregation).” [Online]. Available: https://www.mongodb.com/docs/manual/reference/operator/aggregation/lookup/.

MongoDB Manual, “Aggregation Pipeline Optimization.” [Online]. Available: https://www.mongodb.com/docs/manual/core/aggregation-pipeline-optimization/.

MongoDB Manual, “Indexes.” [Online]. Available: https://www.mongodb.com/docs/manual/indexes/.

MongoDB Manual, “Unique Indexes.” [Online]. Available: https://www.mongodb.com/docs/manual/core/index-unique/.

MongoDB là gì? Tính năng nổi bật và ứng dụng của MongoDB trong thực tế:

https://vietnix.vn/mongodb-la-gi/?utm\_source=ggads&utm\_medium=pmax&gad\_source=1&gad\_campaignid=23376051822&gbraid=0AAAAABwedNLa-KRkPDo6gPkc8rn-nNd6k&gclid=CjwKCAiAmKnKBhBrEiwAaqAnZ54ddVHSkRrgQF\_mPsepzj-G9mh2r5\_TdKPZX6QM-OgJpDvm8WKJJxoCvlIQAvD\_BwE

MongoDB Official. MongoDB Features — Database Capabilities Overview:

https://www.mongodb.com/resources/products/capabilities/features

Viblo. Tìm hiểu về MongoDB:  
https://viblo.asia/p/tim-hieu-ve-mongodb-4P856ajGlY3

Hybrid Technologies. Ưu và nhược điểm của MongoDB:  
https://jobs.hybrid-technologies.vn/blog/uu-nhuoc-diem-mongodb/

Interdata. MongoDB là gì và tại sao được sử dụng rộng rãi?:

https://interdata.vn/blog/mongodb-la-gi/

Làm quen và tổng quan cơ bản về Selenium:  
https://topdev.vn/blog/lam-quen-tom-tat-co-ban-ve-selenium/?utm\_source=google&utm\_medium=cpc&gad\_source=1&gad\_campaignid=23328517423&gbraid=0AAAAADDtBSDOSREUTMcMW1Rt\_dJQhwGm9&gclid=CjwKCAiAmKnKBhBrEiwAaqAnZ3rESnFCtuKfHVZkI6wPxJgxXrZGrN19azDZnjqpK3vWOuBNBiZ11xoCPswQAvD\_BwE

Giới thiệu về Selenium – Bài 1:  
https://viblo.asia/p/bai-1-gioi-thieu-ve-selenium-aWj538VwK6m

Tổng quan về Selenium và vai trò các thành phần:  
https://tech.cybozu.vn/tong-quan-ve-selenium-va-vai-tro-cua-cac-thanh-phan-74a12/

Selenium là gì? Ứng dụng trong kiểm thử tự động:  
https://vietnix.vn/selenium-la-gi/?utm\_source=ggads&utm\_medium=pmax&gad\_source=1&gad\_campaignid=23379656783&gbraid=0AAAAABwedNLs\_9WNHvjFzp7l0Dn1K9FMW&gclid=CjwKCAiAmKnKBhBrEiwAaqAnZ9BYTjtUgUkPSYpvH2WFiHxIgl9D3PdFpuNWo78uDTe4ATzo3AsL2hoC6CcQAvD\_BwE

Selenium Automation Testing – Pros & Cons:

https://testsigma.com/blog/selenium-automation-testing-pros-cons/

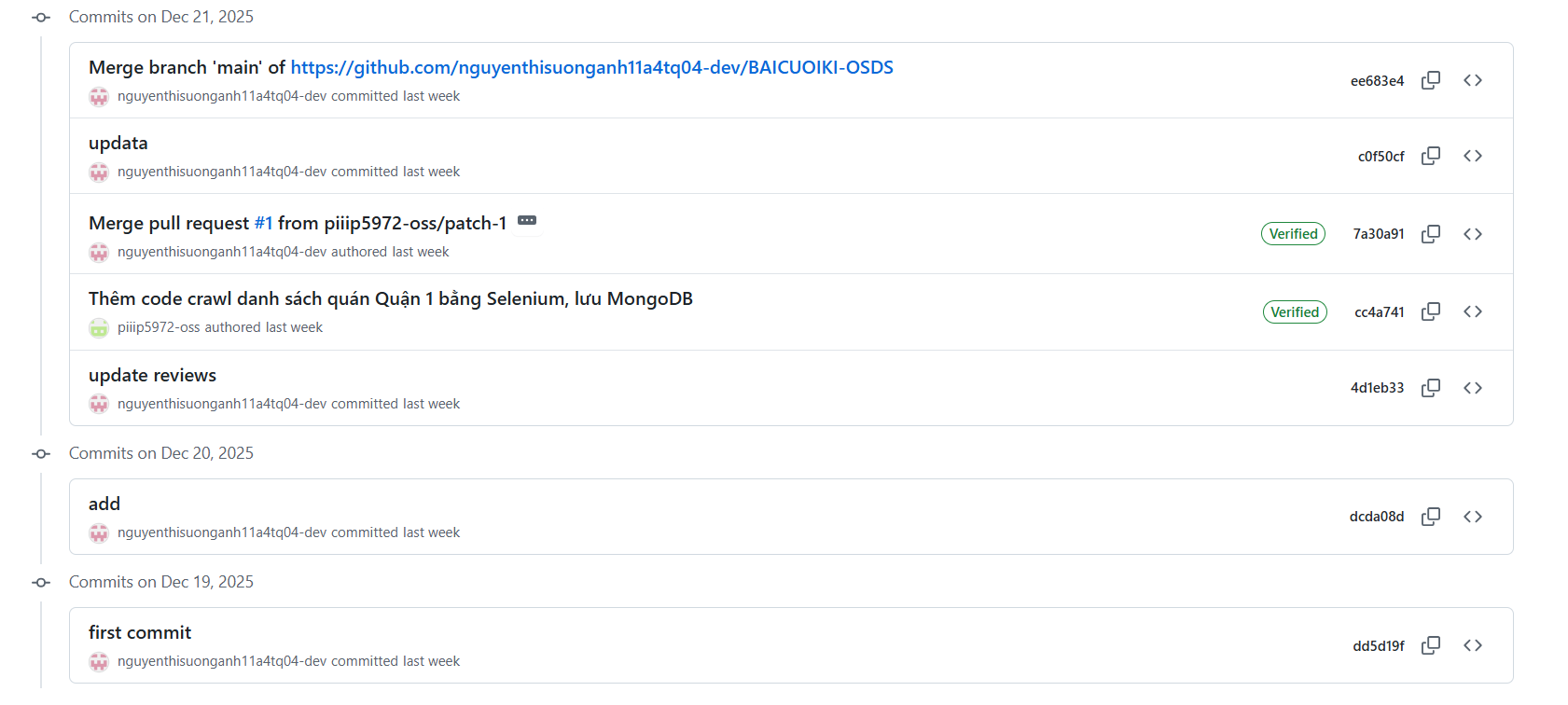
Web Scraping với Selenium Python – Hướng dẫn cơ bản:  
 https://www.scrapingbee.com/blog/selenium-python/

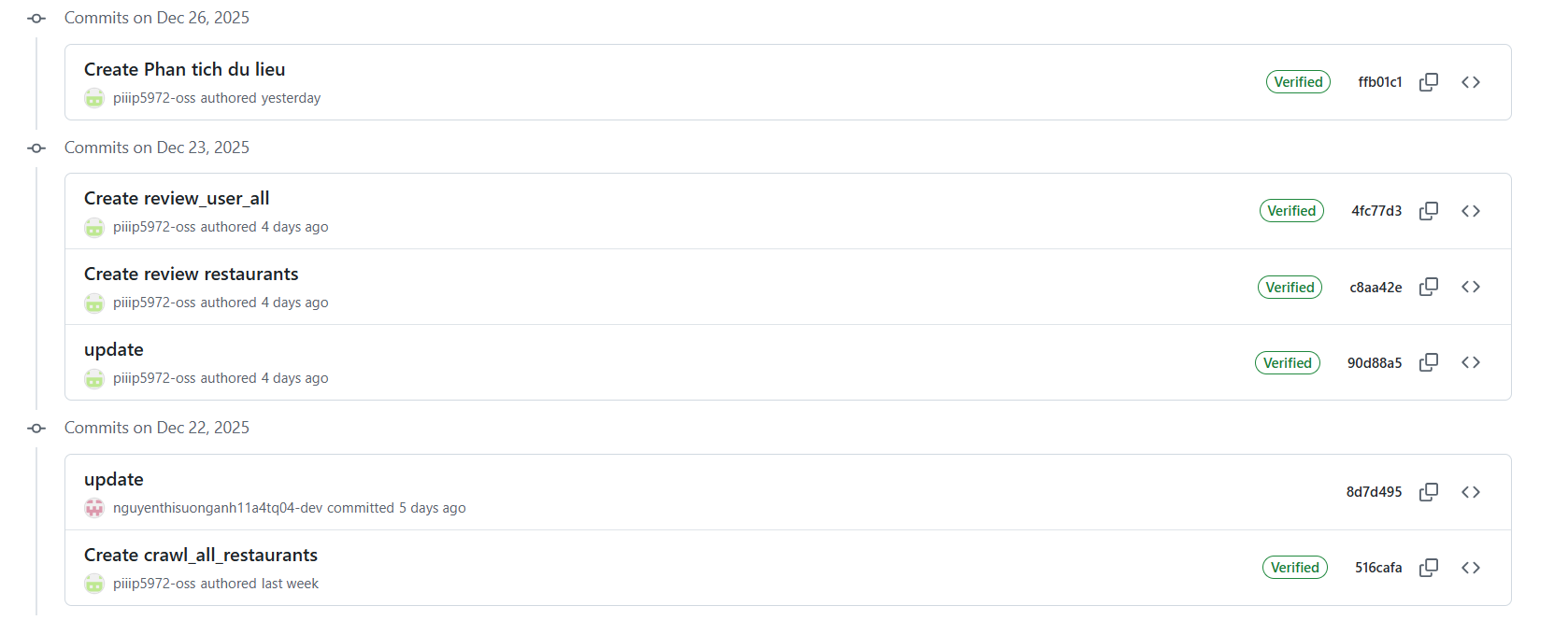
Web Scraping là gì? Kiến thức cần biết:  
https://bkhost.vn/blog/web-scraping/

Web Scraping là gì? Tổng quan và ứng dụng:

https://www.bacs.vn/vi/blog/kien-thuc/web-scraping-la-gi-tat-ca-nhung-gi-ban-can-biet-ve-web-scraping-33766.html?srsltid=AfmBOoqgNSoyFgWJy4mn1T9139Q4hyTeF7ejfg8jvFJyfNjBmG8vIBGZ

# **PHỤ LỤC**





Link dự án: https://github.com/nguyenthisuonganh11a4tq04-dev/BAICUOIKI-OSDS.git