

KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐO ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2024-2025
**TÌM HIỂU GIẢI PHÁP THU THẬP
DỮ LIỆU THÔNG QUA API CỦA CÁC
NỀN TẢNG THƯƠNG MẠI ĐIỆN TỬ**

Giảng viên hướng dẫn:
ThS. Nguyễn Khắc Quốc

Sinh viên thực hiện:
Họ tên: Nguyễn Tân Lộc
MSSV: 110121189
Lớp: DA21TTB

Trà Vinh, tháng 01 năm 2025

KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐO ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2024-2025
**TÌM HIỂU GIẢI PHÁP THU THẬP
DỮ LIỆU THÔNG QUA API CỦA CÁC
NỀN TẢNG THƯƠNG MẠI ĐIỆN TỬ**

Giảng viên hướng dẫn:
ThS. Nguyễn Khắc Quốc

Sinh viên thực hiện:
Họ tên: Nguyễn Tân Lộc
MSSV: 110121189
Lớp: DA21TTB

Trà Vinh, tháng 01 năm 2025

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Trà Vinh, ngày tháng năm
Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

Trà Vinh, ngày tháng năm
Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Em xin chân thành cảm ơn thầy Nguyễn Khắc Quốc đã dành thời gian hướng dẫn tạo điều kiện và giúp em có thêm kiến thức trong quá trình thực hiện đề tài.

Thông qua quá trình thực hiện đề tài, em đã phần nào tích lũy được những kiến thức về API và nền tảng thương mại điện tử, hiểu biết thêm về các giải pháp thu thập dữ liệu, một số phần mềm kiểm tra API và hiểu thêm về các ngôn ngữ lập trình hỗ trợ như Python, Java,...

Trong quá trình thực hiện nghiên cứu đồ án, dù đã cố gắng hết sức nhưng em vẫn không tránh khỏi những sai sót, mong được thầy, cô góp ý và thông cảm. Giúp em học hỏi và cải thiện các dự án trong tương lai.

MỤC LỤC

MỞ ĐẦU	12
CHƯƠNG 1: TỔNG QUAN.....	13
1.1 Application Programming Interface	13
1.2 Thương mại điện tử.....	13
1.3 Phương pháp thu thập và xử lý dữ liệu	14
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT	15
2.1 Tổng quan về API.....	15
2.1.1 <i>Cách thức hoạt động của API</i>	15
2.1.2 <i>Các thành phần chính của API</i>	16
2.1.3 <i>Thành phần chính khi xây dựng API</i>	16
2.1.4 <i>Các loại API</i>	17
2.2 Giải pháp thu thập dữ liệu thông qua API trên các nền tảng TMĐT	18
2.2.1 <i>Khái niệm giải pháp thu thập dữ liệu thông qua API</i>	18
2.2.2 <i>Dữ liệu trong TMĐT</i>	19
2.3 Các công cụ và ngôn ngữ dùng trong quá trình thu thập dữ liệu	21
2.3.1 <i>Ngôn ngữ</i>	21
2.3.2 <i>Công cụ</i>	22
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU.....	24
3.1 Lựa chọn nền tảng TMĐT	24
3.1.1 <i>Một số nền tảng TMĐT</i>	24

3.1.2 Tài liệu API của một số nền tảng TMĐT.....	25
3.2 Sử dụng công cụ Postman kiểm thử API	26
3.2.1 Các thành phần trong Postman	26
3.2.2 Làm việc với GET request	28
3.2.3 Làm việc với POST request	28
3.3 Giải pháp thu thập dữ liệu API bằng Python	31
3.3.1 Lấy đường dẫn API và kiểm tra.....	31
3.3.2 Sử dụng Python để thu thập dữ liệu.....	34
3.4 Giải pháp thu thập dữ liệu thông qua phần mềm quản lý đám mây.....	37
3.4.1 Một số phần mềm quản lý đám mây hỗ trợ thu thập API.....	37
3.4.2 Quy trình thực hiện.....	38
3.5 Giải pháp thu thập dữ liệu thông qua việc bán hàng	42
3.6 Giải pháp thu thập dữ liệu thông qua việc hợp tác với các cửa hàng.....	44
3.6.1 Đăng ký ứng dụng khách hàng	44
3.6.2 Sử dụng Postman để nhận token.....	48
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU.....	55
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	59

DANH MỤC HÌNH ẢNH

Hình 1 Cách thức hoạt động của API.....	15
Hình 2 Cổng API (API Gateway)	16
Hình 3 Định dạng JSON	20
Hình 4 Định dạng CSV	20
Hình 5 Định dạng XML	21
Hình 6 Ngôn ngữ lập trình JavaScript.....	21
Hình 7 Ngôn ngữ lập trình Python	22
Hình 8 Công cụ kiểm thử Tricentis Tosca	22
Hình 9 Công cụ kiểm thử Postman	23
Hình 10 Công cụ thu thập dữ liệu cURL	23
Hình 11 Công cụ thu thập dữ liệu Visual Studio	23
Hình 12 Nền tảng TMĐT Tiki	25
Hình 13 Nền tảng TMĐT Amazon	25
Hình 14 Các thành phần trong Postman.....	26
Hình 15 GET request	28
Hình 16 Tạo request mới.....	29
Hình 17 Tab mới	29
Hình 18 Trong body	29
Hình 19 Sao chép và dán kết quả	30
Hình 20 Kết quả trả về	31
Hình 21 Nền tảng TMĐT Tiki	32
Hình 22 Kiểm tra lấy đường dẫn API	32
Hình 23 Kiểm tra nội dung đường dẫn	33
Hình 24 Truy cập Postman.....	33
Hình 25 Nội dung kiểm tra.....	33
Hình 26 Không lấy được dữ liệu.....	34
Hình 27 Lấy được dữ liệu	34
Hình 28 Sử dụng Google Colab	34
Hình 29 Lập trình lấy dữ liệu API	35
Hình 30 Kết quả trả về	37

Hình 31 Kết quả được sắp xếp ở dạng bảng	37
Hình 32 Cấu hình API.....	39
Hình 33 Tạo tài nguyên API	40
Hình 34 Lưu trữ API trong cơ sở dữ liệu.....	41
Hình 35 Chuyển đổi dữ liệu	41
Hình 36 Xuất dữ liệu.....	42
Hình 37 Đăng ký hoặc đăng nhập tài khoản bán hàng.....	42
Hình 38 Điện thông tin đăng ký	42
Hình 39 Trang chủ	43
Hình 40 Danh sách sản phẩm.....	43
Hình 41 Tạo sản phẩm	43
Hình 42 Quản lý đánh giá	44
Hình 43 Trang tạo ứng dụng	45
Hình 44 Chọn phương thức.....	46
Hình 45 Ứng dụng công cộng	46
Hình 46 Ứng dụng nội bộ.....	47
Hình 47 Xác nhận kết nối.....	48
Hình 48 Tạo yêu cầu Postman mới	48
Hình 49 Kết quả	49
Hình 50 Đăng nhập	50
Hình 51 Trao quyền cho ứng dụng	51
Hình 52 Mã thông báo truy cập.....	52
Hình 53 Tạo yêu cầu Postman	52
Hình 54 Kết quả	53
Hình 55 Mã truy cập	54

DANH MỤC TỪ VIẾT TẮT

STT	Ký tự viết tắt	Chữ viết đầy đủ
1	TMĐT	Thương mại điện tử
2	API	Application Programming Interface
3	URL	Uniform Resource Locator

TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

Dữ liệu là phần quan trọng không thể thiếu của các nền tảng TMĐT. Việc tìm hiểu giải pháp thu thập dữ liệu thông qua API từ các nền tảng TMĐT giúp cho các doanh nghiệp đưa ra những quyết định và chiến lược kinh doanh hiệu quả.

Giải pháp là quá trình tìm ra được phương án hoặc cách thức thực hiện để thông qua API thu thập dữ liệu từ các nền tảng TMĐT. Yêu cầu của giải pháp bao gồm phương pháp hiệu quả để thu thập dữ liệu, quy trình thực hiện, cách thức hoạt động và xử lý của API.

Thu thập dữ liệu là quá trình lấy dữ liệu thông qua API từ các nền tảng TMĐT để tổng hợp, phân tích và xử lý các dữ liệu đã thu thập, từ đó hỗ trợ cho các hoạt động kinh doanh như: phân tích thị trường, tối ưu hóa các hoạt động kinh doanh, nghiên cứu và phát triển sản phẩm,...giúp phục vụ cho quá trình kinh doanh hiệu quả.

Các hướng tiếp cận

Xác định rõ mục tiêu thu thập dữ liệu và Chọn nền tảng TMĐT

Dữ liệu cần thu thập: sản phẩm, tên sản phẩm, giá sản phẩm, lượt mua, đơn hàng, tương tác đánh giá,..

Mục đích sử dụng: xử lý, thống kê, phân tích, báo cáo,...

Mỗi nền tảng đều có API riêng với các đặc điểm khác nhau, một số nền tảng hiện nay như: Amazon, Alibaba, Shopee, Lazada, Tiki,...

Tìm hiểu API của nền tảng và chọn công cụ, ngôn ngữ lập trình

Tài liệu API: hiểu rõ cách sử dụng của API, các điểm cuối, phương thức, định dạng dữ liệu.

Giới hạn API: số lượng yêu cầu, số lần truy cập, lượng dữ liệu.

Công cụ: Python (Requests, BeautifulSoup, Scrapy),...

Ngôn ngữ: Python, JavaScript,...

Thu thập và phân tích, xử lý dữ liệu

Gửi yêu cầu đến API, xử lý dữ liệu trả về và lưu trữ vào hệ thống.

Loại bỏ những dữ liệu sai, trùng lặp, thống kê mô tả dữ liệu, biểu đồ dữ liệu.

Cách giải quyết vấn đề

Trong quá trình thu thập dữ liệu, có thể gặp một số vấn đề như:

Không xác định rõ những loại dữ liệu cần thu thập dẫn đến thiếu thông tin hoặc bị trùng lặp.

Số lượng yêu cầu API bị giới hạn có thể dẫn đến dữ liệu thu thập không đầy đủ.

Quá nhiều dữ liệu dẫn đến bị chậm hoặc gây thiếu sót dữ liệu trong quá trình xử lý.

Để giải quyết những vấn đề trên, cần xác định rõ yêu cầu thu thập, tập trung vào các thông tin phục vụ trực tiếp cho mục đích sử dụng, tối ưu hóa số lần thu thập. Sắp xếp, lưu trữ dữ liệu theo từng phần hoặc từng mục để thuận lợi cho việc phân tích và xử lý.

Một số kết quả đạt được

Giải pháp thu thập dữ liệu thông qua API mang lại nhiều lợi ích cho các doanh nghiệp, cửa hàng,... trên các nền tảng TMĐT.

Tăng hiệu quả kinh doanh: tối ưu hóa các quyết định kinh doanh, phát triển sản phẩm mới và những dịch vụ phù hợp với nhu cầu của khách hàng. Dựa vào dữ liệu thu thập thông qua API cung cấp cho các hoạt động thống kê, phân tích, báo cáo.

Tăng khả năng cạnh tranh: thu thập dữ liệu thông qua API trên các nền tảng TMĐT giúp các cửa hàng, doanh nghiệp đưa ra những quyết định chính xác, nâng cao khả năng cạnh tranh. Dựa vào việc thu thập những mảnh hàng được bán ra, lượt mua, đánh giá sản phẩm,...

Tăng sự hài lòng, yêu thích của khách hàng: thu thập dữ liệu tốt và hiệu quả giúp các cửa hàng, doanh nghiệp hiểu rõ hơn những vấn đề mà khách hàng gặp phải khi sử dụng sản phẩm, dịch vụ để đưa ra hướng giải quyết phù hợp, làm cho khách hàng hài lòng, yêu thích và sẽ ưu tiên chọn cửa hàng, doanh nghiệp khi mua hàng hoặc sử dụng dịch vụ. Dựa vào việc thu thập dữ liệu đánh giá, phản hồi nhận xét của khách hàng.

Giải pháp thu thập dữ liệu thông qua API trên các nền tảng TMĐT là phương pháp cần thiết trong việc thu thập dữ liệu. Giải pháp thu thập dữ liệu thông qua API sẽ giúp cho các cửa hàng doanh nghiệp nắm bắt được những nhu cầu của khách hàng, để đưa ra những quyết định, hướng phát triển, chiến lược kinh doanh phù hợp mang lại hiệu quả cao cho các cửa hàng và doanh nghiệp.

MỞ ĐẦU

1. Lý do chọn đề tài

- Với sự phát triển về công nghệ hóa hiện nay, TMĐT đã trở thành một phần không thể thiếu trong cuộc sống hằng ngày, đặc biệt là trong lĩnh vực kinh tế.
- Việc thu thập dữ liệu để nắm bắt và phân tích nhu cầu của người tiêu dùng từ các nền tảng TMĐT giúp cho các doanh nghiệp có cái nhìn tổng quan và kịp thời về thị trường để đưa ra những chiến lược kinh doanh phù hợp.
- API là công cụ hiệu quả để kết nối và trao đổi dữ liệu giữa các hệ thống. Việc sử dụng API để thu thập dữ liệu giúp tiết kiệm thời gian và công sức.

2. Mục đích

- Hiểu rõ về API và giải pháp thu thập dữ liệu: nghiên cứu về API, cách thức hoạt động, giải pháp thu thập và khả năng phát triển trong lĩnh vực TMĐT
- Xây dựng giải pháp: xây dựng các bước của giải pháp thu thập dữ liệu, gồm lựa chọn API phù hợp, thiết kế cấu trúc dữ liệu, triển khai và vận hành.
- Đánh giá hiệu quả: đánh giá giải pháp thu dữ liệu, gồm tốc độ thu tập dữ liệu, độ chính xác và ổn định.

3. Đối tượng nghiên cứu

- API của các nền tảng TMĐT: nghiên cứu API được cung cấp từ các nền tảng TMĐT phổ biến như Shopee, Amazon, Lazada,..
- Dữ liệu sản phẩm: thông tin, tên sản phẩm, mức giá, mô tả, hình ảnh, đánh giá của khách hàng.
- Dữ liệu người dùng: thông tin, lịch sử mua hàng, thông tin tìm kiếm, đánh giá về sản phẩm.

4. Phạm vi nghiên cứu

- Nghiên cứu lý thuyết: nghiên cứu các khái niệm, tài liệu API, thương mại điện tử, các giải pháp thu thập dữ liệu.
- Thủ nghiệm: thử nghiệm các giải pháp thu thập dữ liệu thông qua API của các nền tảng TMĐT.
- Phân tích và đánh giá: phân tích và đánh giá dữ liệu đã được thu thập từ các nền tảng TMĐT.

CHƯƠNG 1: TỔNG QUAN

1.1 Application Programming Interface

1.1.1 Khái niệm API

- API là viết tắt của Application Programming Interface - giao diện lập trình ứng dụng đây là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác.
- API là phương tiện cho hai hay nhiều ứng dụng có thể trao đổi, tương tác với nhau, giúp tạo ra tương tác giữa người dùng với ứng dụng một cách tiện lợi và hiệu quả hơn.

1.1.2 Đặc điểm nổi bật của API

- Có khả năng tự động hóa sản phẩm
- Tích hợp vô cùng linh động và dễ dàng
- Thông tin thời gian được cập nhật linh hoạt
- Tiêu chuẩn chung giúp việc sử dụng cực kỳ dễ dàng

1.2 Thương mại điện tử

1.2.1 Khái niệm TMĐT

- TMĐT còn được gọi là (Ecommerce/electronic commerce) là hoạt động mua, bán, trao đổi, hàng hóa và dịch vụ thông qua các hệ thống điện tử như Internet và mạng máy tính.

- Thương mại điện tử còn được biết đến với nhiều tên gọi khác nhau:
 - Thương mại trực tuyến (online trade)
 - Thương mại điều khiển học (cyber trade)
 - Thương mại không giấy tờ (paperless commerce)
 - Thương mại Internet (Internet commerce)
 - Thương mại số hóa (Digital commerce)
 - Kinh doanh điện tử (E-business)

1.2.2 Các loại hình giao dịch TMĐT

- B2C (Business to Customer): Giao dịch giữa doanh nghiệp với người tiêu dùng.
- B2B (Business to Business): Giao dịch giữa doanh nghiệp với doanh nghiệp.
- B2G (Business to Government): Giao dịch giữa doanh nghiệp với chính phủ.
- C2C (Consumer to Consumer): Giao dịch giữa người tiêu dùng với người tiêu dùng
- C2G (Consumer to Government): Giao dịch giữa người dân với chính phủ.

- G2C (Government to Consumer): Giao dịch giữa chính phủ với người dân.
- G2G (Government to Government): Giao dịch giữa các cơ quan chính phủ
- B2E (Business to Employee) Giao dịch trong nội bộ doanh nghiệp.

1.3 Phương pháp thu thập và xử lý dữ liệu

1.3.1 Xác định dữ liệu cần thu thập:

- Dữ liệu sản phẩm: tên sản phẩm, mô tả, giá cả, hình ảnh, danh mục, thuộc tính, tồn kho,...
- Dữ liệu đơn hàng: mã đơn hàng, khách hàng, sản phẩm, số lượng, tổng tiền, trạng thái đơn hàng,...
- Dữ liệu khách hàng: thông tin cá nhân, lịch sử mua hàng, hành vi mua sắm,...

1.3.2 Chọn API phù hợp:

- Tìm hiểu tài liệu API: mỗi nền tảng TMĐT sẽ có tài liệu API riêng, cung cấp thông tin chi tiết về các endpoint, phương thức gọi API, định dạng dữ liệu trả về,...
- Xác định các giới hạn: một số API có giới hạn về số lượng yêu cầu, tần suất truy cập, hoặc lượng dữ liệu trả về.

1.3.3 Thiết kế cấu trúc dữ liệu:

- Xây dựng schema: xác định cách dữ liệu sẽ được lưu trữ trong cơ sở dữ liệu hoặc kho dữ liệu.
- Chuẩn hóa dữ liệu: đảm bảo dữ liệu có cấu trúc nhất quán và dễ dàng xử lý.

1.3.4 Xử lý và phân tích dữ liệu:

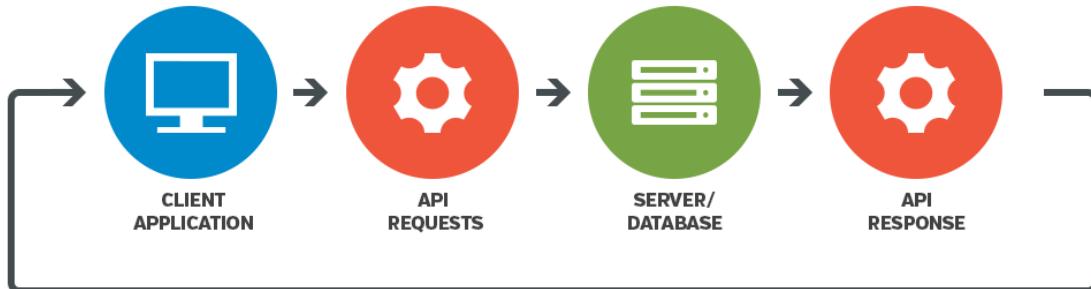
- Sàn lọc dữ liệu: loại bỏ các dữ liệu trùng lặp, thiếu sót, hoặc sai sót.
- Biến đổi dữ liệu: chuyển đổi dữ liệu sang dạng phù hợp với các phép tính thống kê hoặc xây dựng mô hình.
- Phân tích dữ liệu: sử dụng các kỹ thuật thống kê, học máy để trích xuất thông tin hữu ích từ dữ liệu.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

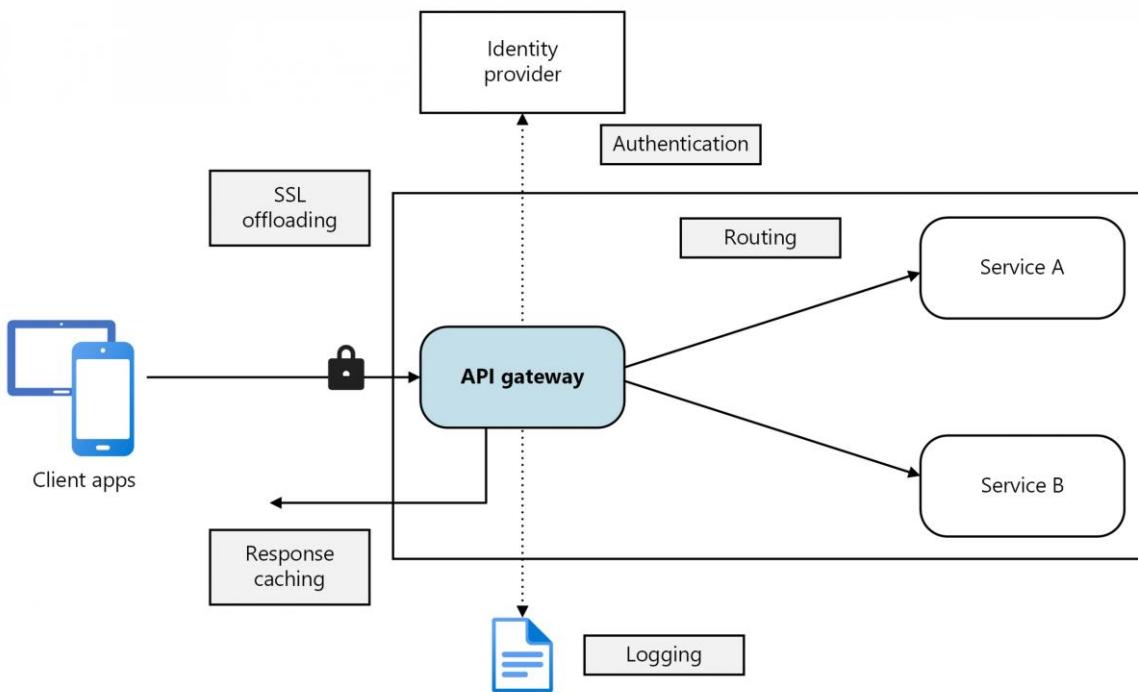
2.1 Tổng quan về API

2.1.1 Cách thức hoạt động của API

- Cách thức hoạt động của API có thể được giải thích rõ ràng bằng một vài bước đơn giản.
- API hoạt động như một phương tiện liên lạc giữa hai chương trình hoặc hệ thống để hoạt động. Máy khách là người dùng/khách hàng (người gửi yêu cầu), phương tiện là API và máy chủ là backend (nơi yêu cầu được chấp nhận và phản hồi được cung cấp).
- Các bước tiếp theo trong quá trình bao gồm:
 - Máy khách bắt đầu các yêu cầu thông qua API URI (Uniform Resource Identifier).
 - API thực hiện cuộc gọi đến máy chủ sau khi nhận được yêu cầu.
 - Sau đó, máy chủ sẽ gửi phản hồi trả lại API cùng với thông tin.
 - Cuối cùng, API chuyển dữ liệu đến máy khách.



Hình 1 Cách thức hoạt động của API



Hình 2 Cổng API (API Gateway)

2.1.2 Các thành phần chính của API

- Endpoint: Đây là URL mà qua đó các yêu cầu (requests) được gửi đến server để xử lý.
- Request: Yêu cầu gửi đến API để thực hiện một hành động cụ thể, chẳng hạn như lấy dữ liệu (GET), tạo mới dữ liệu (POST), hoặc sửa dữ liệu (PUT).
- Response: Phản hồi từ API chứa dữ liệu yêu cầu hoặc kết quả của hành động.
- Authentication: Một số API yêu cầu xác thực người dùng (ví dụ: qua API keys hoặc token) để bảo mật và kiểm soát quyền truy cập.

2.1.3 Thành phần chính khi xây dựng API

- **Request:** Một cái request đúng chuẩn cần có 4 thứ:

1. **URL:** Là địa chỉ duy nhất cho 1 request, thường là đường dẫn tới một hàm xử lí logic.
2. **Method:** HTTP request có tất cả 9 loại method , 2 loại được sử dụng phổ biến nhất là GET và POST
 - GET: Sử dụng để lấy thông tin từ server theo URI đã cung cấp.
 - HEAD: Giống với GET nhưng response trả về không có body, chỉ có header.
 - POST: Gửi thông tin tới server thông qua các parameters HTTP.
 - PUT: Ghi đè tất cả thông tin của đối tượng với những gì được gửi lên.

- PATCH: Ghi đè các thông tin được thay đổi của đối tượng.
 - DELETE: Xóa resource trên server.
 - CONNECT: Thiết lập một kết nối tới server theo URI.
 - OPTIONS: Mô tả các tùy chọn giao tiếp cho resource.
 - TRACE: Thực hiện một bài test loop-back theo đường dẫn đến resource.
3. **Headers:** Là nơi chứa các thông tin cần thiết của 1 request nhưng end-users không biết có sự tồn tại của nó. Ví dụ: độ dài của request body, thời gian gửi request, loại thiết bị đang sử dụng, loại định dạng cái response mà client có đọc được...
4. **Body:** Là nơi chứa thông tin mà client sẽ điền.
- **Response:** Sau khi nhận được request từ phía client, server sẽ xử lý cái request đó và gửi ngược lại cho client 1 cái response. Cấu trúc của 1 response tương đối giống phần request nhưng Status code sẽ thay thế cho URL và Method. Tóm lại, nó có cấu trúc 3 phần:
- Status code
 - Headers
 - Body
- Phần Header và body tương đối giống với request.

2.1.4 Các loại API

- API về mặt quyền truy cập, được chia thành 4 loại:

- **API mở (open API):** Đây là API công khai. Giống như tên gọi, các API này có sẵn công khai, vậy nên sẽ không có bất kỳ hạn chế nào khi người dùng truy cập các API này.
- **API đối tác (Partner API):** API này không có sẵn công khai, chính vì vậy chỉ khi được cấp quyền hoặc các giấy phép liên quan thì mới có thể truy cập loại API này.
- **API nội bộ (Internal API):** API này còn có tên gọi khác là API riêng tư, chỉ những hệ thống nội bộ mới có quyền truy cập và sử dụng loại API này.
Internal API thường được sử dụng trong phạm vi công ty để cải thiện các sản phẩm, dịch vụ của mình.
- **API tổng hợp:** Đây là sự kết hợp của 2 API khác nhau để giải quyết những vấn đề phức tạp của hệ thống.

- API về mặt kiến trúc, được chia thành 5 loại:

- **REST API:** Cho phép thực hiện các hoạt động CRUD (tạo, đọc, cập nhật và xóa) giữa máy khách và máy chủ. Đồng thời cung cấp nhiều API Endpoint để thao tác dữ liệu.
- **GraphQL API:** Đây là ngôn ngữ truy vấn cho phép máy khách yêu cầu dữ liệu chính xác mà máy khách yêu cầu từ máy chủ. Do đó, loại bỏ các vấn đề tình trạng “over-fetching” và “under-fetching” xảy ra trong các hệ điều hành khác.
- **SOAP API:** Bất kỳ dịch vụ web nào tuân thủ đặc tả API SOAP web service đều là SOAP web service.
- **Kafka API:** Apache Kafka là một nền tảng phát trực tuyến sự kiện kết hợp ba khả năng để có thể triển khai các trường hợp sử dụng khác nhau. 3 khả năng này là xuất bản và đăng ký các luồng sự kiện, lưu trữ các luồng sự kiện lâu dài và đáng tin cậy, đồng thời xử lý các luồng sự kiện khi chúng xảy ra.
- **Async API:** Các API này hoạt động theo nguyên tắc Event-Driven Architecture (EDA). Trong Async API, nhiều Subscriber có thể đăng ký Publisher. Khi một sự kiện xảy ra, Publisher (nhà cung cấp API) có thể thông báo cho tất cả Subscriber thông qua Message broker....

2.2 Giải pháp thu thập dữ liệu thông qua API trên các nền tảng TMĐT

2.2.1 Khái niệm giải pháp thu thập dữ liệu thông qua API

- Giải pháp thu thập dữ liệu thông qua API là một phương pháp lấy thông tin từ các cửa hàng trực tuyến. Có thể truy cập từng trang web và sao chép dữ liệu hay sử dụng các giao diện lập trình ứng dụng (API) mà các nền tảng này cung cấp. API đóng vai trò như một cầu nối, cho phép các ứng dụng khác tương tác và lấy dữ liệu từ hệ thống của nền tảng đó.

- API của các nền tảng TMĐT là công cụ kết nối và cung cấp quyền truy cập vào dữ liệu của họ một cách có kiểm soát.

- Các API thường được thiết kế để hỗ trợ người bán hàng, nhà phát triển, và các đối tác tích hợp công cụ của họ vào nền tảng.

2.2.2 Dữ liệu trong TMĐT

- Các dữ liệu trong TMĐT

+ Dữ liệu khách hàng:

Dữ liệu về khách hàng cung cấp thông tin về khách hàng như: tên, địa chỉ, số điện thoại, email,.. thói quen mua sắm, sở thích và nhu cầu mua hàng.

+ Dữ liệu cửa hàng

Dữ liệu về cửa hàng cung cấp thông tin về cửa hàng như: địa chỉ cửa hàng giờ mở cửa, thông tin liên hệ,..

+ Dữ liệu sản phẩm:

Dữ liệu về sản phẩm bao gồm: loại sản phẩm, mô tả sản phẩm, giá cả, đánh giá của khách hàng, danh giá của sản phẩm, số lượng bán hàng và tần suất mua hàng.

+ Dữ liệu về doanh thu

Dữ liệu về doanh thu giúp đánh giá hiệu quả kinh doanh, mức độ ứng dụng của các chiến lược và chính sách bán hàng.

+ Dữ liệu đánh giá

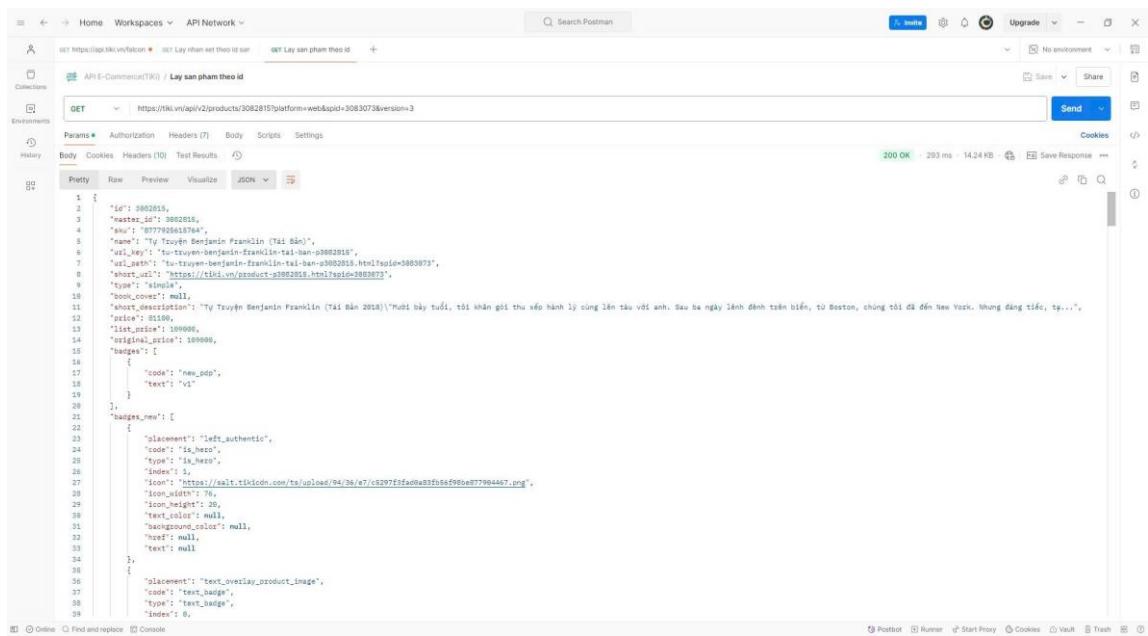
Dữ liệu đánh giá về sản phẩm là những thông tin mà khách hàng cung cấp sau khi trải nghiệm sản phẩm hoặc dịch vụ. Là một nguồn dữ liệu quý giá, giúp doanh nghiệp hiểu rõ hơn về chất lượng sản phẩm, nhu cầu của khách hàng và từ đó đưa ra những quyết định kinh doanh hiệu quả.

- Định dạng dữ liệu phổ biến

+ JSON

JSON là viết tắt của Javascript Object Notation, là định dạng phổ biến nhất cho các lệnh gọi API, định dạng dữ liệu có thể được sử dụng độc lập với Javascript và hầu hết ngôn ngữ lập trình hiện nay đều có thể đọc (phân tích cú pháp) cũng như tạo mã JSON.

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử



The screenshot shows the Postman application interface. A GET request is made to <https://tiki.vn/api/v2/products/3082815?platform=web&spid=3083073&version=3>. The response status is 200 OK, with a response time of 293 ms and a size of 14.24 KB. The response body is a JSON object representing a product:

```
1 {
2   "id": 3082815,
3   "ext_id": 3082815,
4   "sku": "8777926415744",
5   "name": "Tý Truyện Benjamin Franklin (Tái Bản 2018)",
6   "url": "https://tiki.vn/tuy-truyen-benjamin-franklin-tai-ban-p3082815",
7   "url_latin": "https://tiki.vn/tuy-truyen-benjamin-franklin-tai-ban-p3082815.html?spid=3083073",
8   "short_url": "https://tiki.vn/product-p3082815.html?spid=3083073",
9   "type": "simple",
10   "book_cover": null,
11   "long_desc": "Tý Truyện Benjamin Franklin (Tái Bản 2018) \"Mười ngày tuổi, tôi nhận gói thư xác nhận lý cùng tên tàu với anh. Sau ba ngày lặn đánh trên biển, từ Boston, chúng tôi đã đến New York. Nhưng đáng tiếc, ta...\"",
12   "price": 81000,
13   "list_price": 109000,
14   "original_price": 109000,
15   "badges": [
16     {
17       "code": "new_pop",
18       "text": "v1"
19     }
20   ],
21   "badges_new": [
22     {
23       "placement": "left_authentic",
24       "code": "tk_hexo",
25       "type": "hexo",
26       "index": 1,
27       "icon": "https://salt.tikicon.com/ta/upload/94/36/e7/c5297f7fad9a83fb8ef98e877984a67.png",
28       "icon_width": 76,
29       "icon_height": 29,
30       "text_color": null,
31       "background_color": null,
32       "size": null,
33       "text": null
34     },
35     {
36       "placement": "text_overlay_product_image",
37       "code": "text_badge",
38       "type": "text_badge",
39       "index": 0
40     }
41   ]
42 }
```

Hình 3 Định dạng JSON

+ CSV

CSV là định dạng lưu trữ dữ liệu lưu trữ các giá trị dữ liệu (văn bản thuần túy) theo định dạng danh sách được phân tách bằng dấu phẩy. Bảng tính hoặc cơ sở dữ liệu quan hệ có thể dễ dàng truy cập tệp CSV trong khi cung cấp nhiều tính năng chức năng trong phân tích và thao tác dữ liệu.

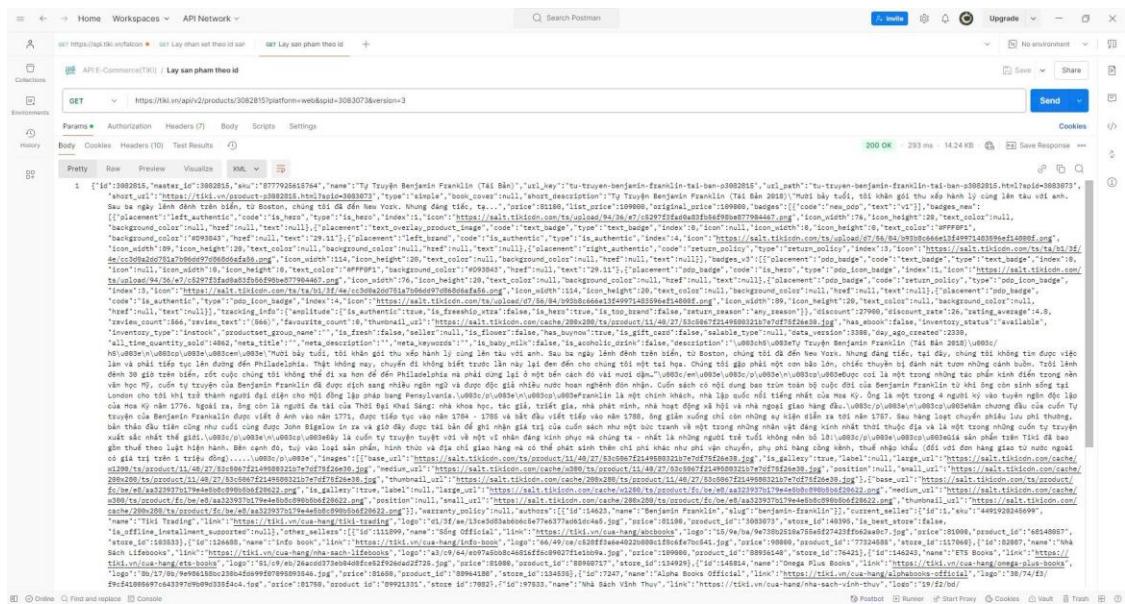
```
ticker,date,open,high,low,close,adj close,volume
AAPL,2021-02-02,135.73,136.3,134.61,134.99,134.359,79426446
AAPL,2021-02-01,133.75,135.38,130.932,134.14,133.513,104212352
AAPL,2021-01-29,135.83,136.74,130.21,131.96,131.343,172209910
AAPL,2021-01-28,139.52,141.99,136.7,137.09,136.449,137245542
AAPL,2021-01-27,143.43,144.3,140.41,142.06,141.396,121162513
```

Hình 4 Định dạng CSV

+ XML

XML là định dạng trao đổi dữ liệu cho API trước JSON. Đây là ngôn ngữ đánh dấu mà cả con người và máy đều có thể đọc được. Mặc dù nó không nhỏ gọn và được tối ưu hóa cho tốc độ đọc. Nó được coi là dài dòng và dư thừa khi so sánh với JSON. Mặc dù XML phù hợp hơn để kết hợp các tập thông tin từ các hệ thống khác nhau như siêu dữ liệu.

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử

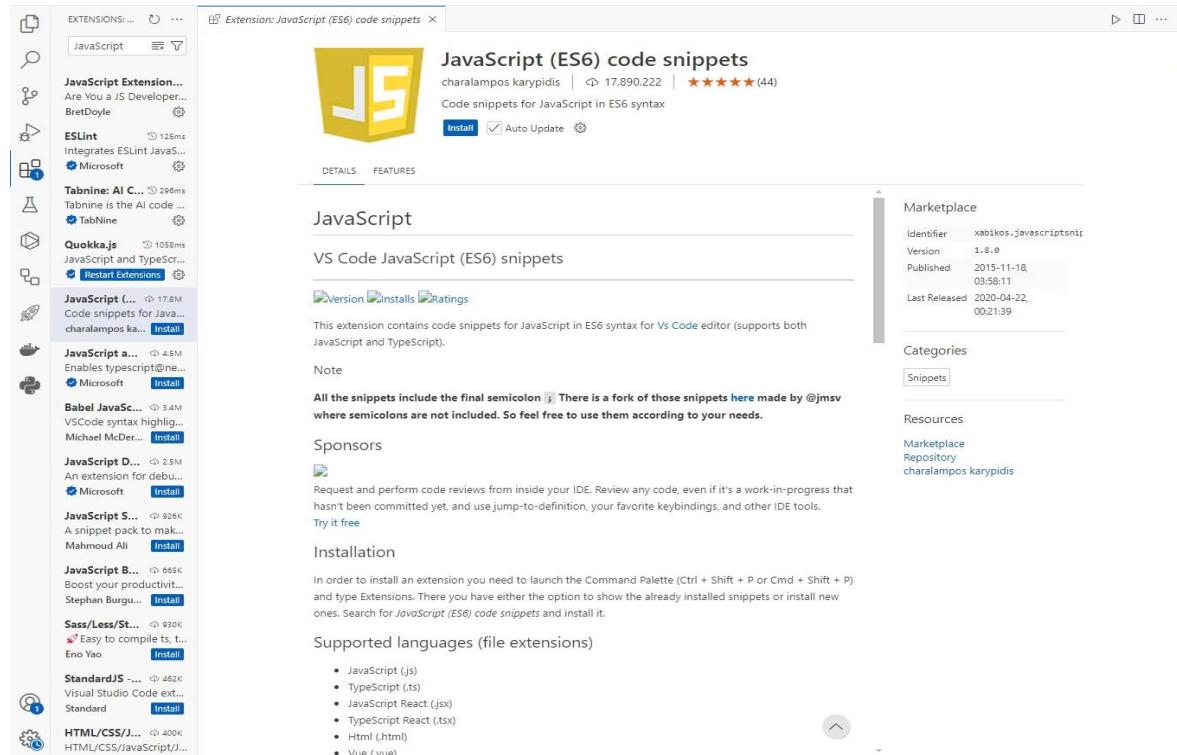


Hình 5 Định dạng XML

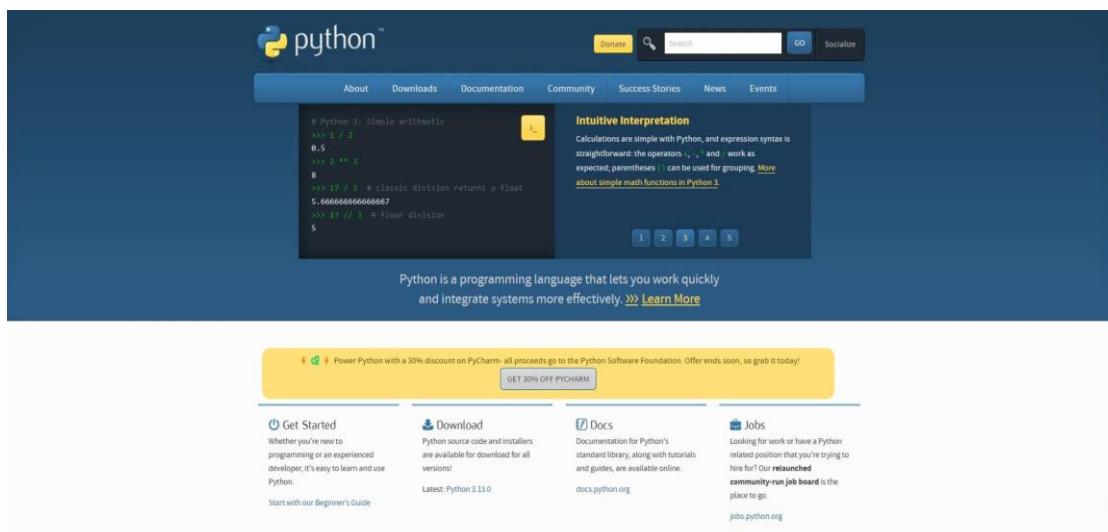
2.3 Các công cụ và ngôn ngữ dùng trong quá trình thu thập dữ liệu

2.3.1 Ngôn ngữ

- Ngôn ngữ lập trình đóng vai trò quan trọng trong việc thu thập dữ liệu thông qua API. Để kết nối với API, xử lý và tận dụng dữ liệu hiệu quả, đồng thời giúp mở rộng khả năng ứng dụng của dữ liệu trong các hệ thống khác nhau.
- Những ngôn ngữ lập trình sử dụng trong việc thu thập dữ liệu như: Python, Java, C#, JavaScript,...



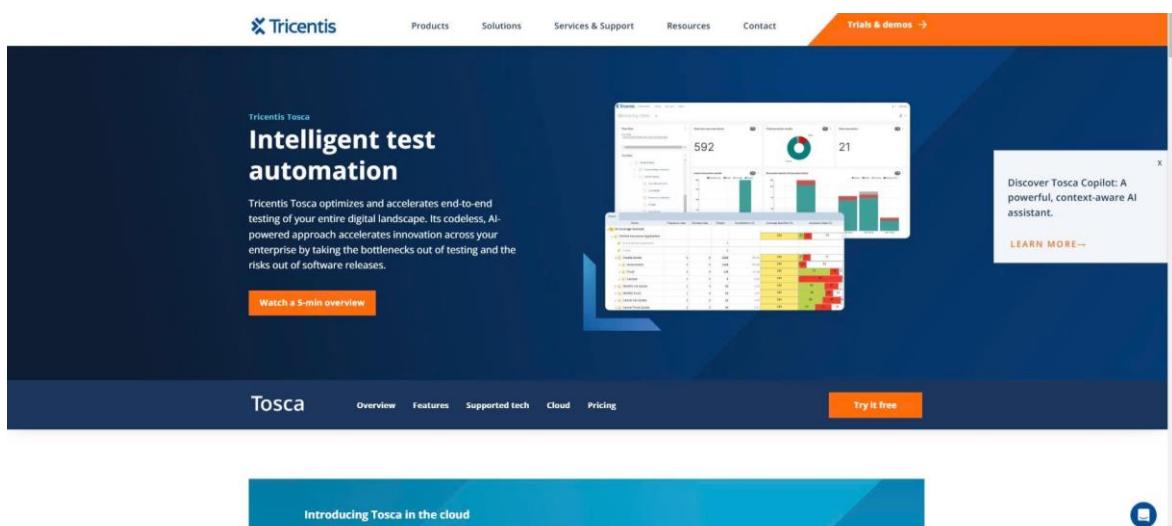
Hình 6 Ngôn ngữ lập trình JavaScript



Hình 7 Ngôn ngữ lập trình Python

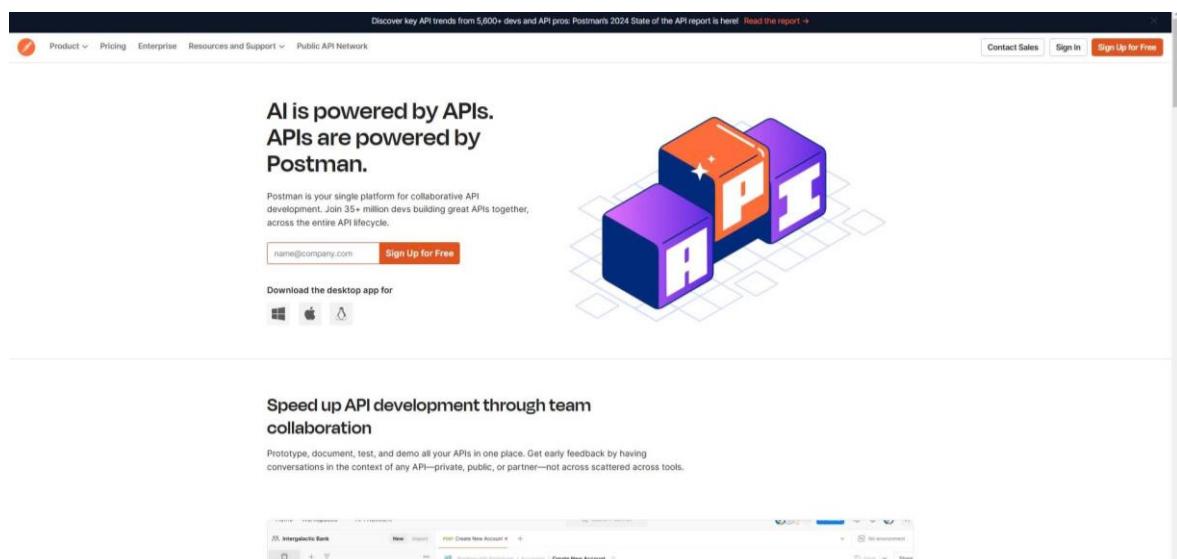
2.3.2 Công cụ

- Các công cụ hỗ trợ cho việc thu thập dữ liệu, kiểm tra quá trình đánh giá và xác minh xem đường dẫn API trên các nền tảng TMĐT có thể sử dụng được hay không, thống kê và phân tích dữ liệu đã xác minh. Các công cụ đóng vai trò không thể thiếu trong việc thu thập dữ liệu giúp tăng khả năng thu thập, giảm thiểu dữ liệu bị sai và thiếu sót...
- Những công cụ dùng để kiểm thử như: Postman, Jmeter, Katalon Studio, Tricentis Tosca, Apigee,...



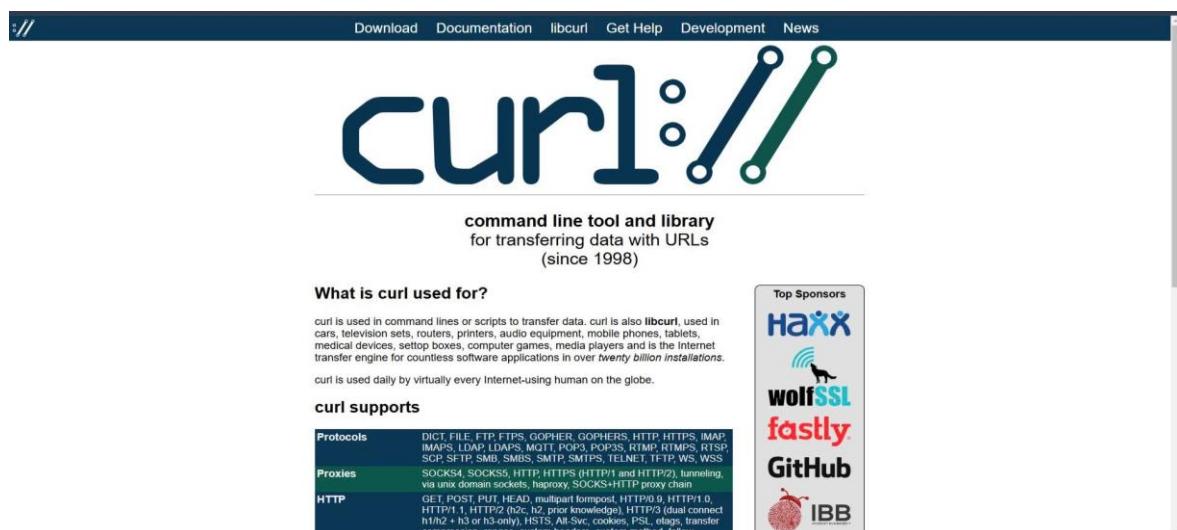
Hình 8 Công cụ kiểm thử Tricentis Tosca

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử



Hình 9 Công cụ kiểm thử Postman

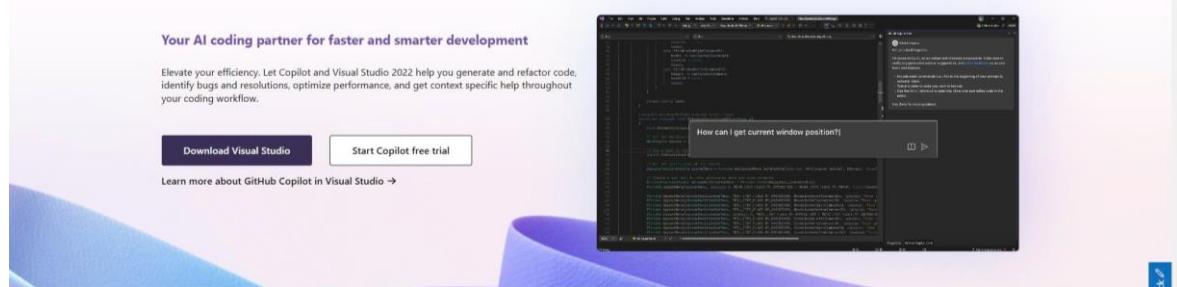
- Công cụ dùng để thu thập dữ liệu như: cURL, Visual Studio, Eclipse,...



Hình 10 Công cụ thu thập dữ liệu cURL



GitHub Copilot woven into Visual Studio



Meet the Visual Studio family

Hình 11 Công cụ thu thập dữ liệu Visual Studio

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Lựa chọn nền tảng TMĐT

- Nền tảng thương mại điện tử là phần mềm cho phép các doanh nghiệp tạo ra các cửa hàng trực tuyến. Trong các cửa hàng trực tuyến này, các doanh nghiệp có thể bán sản phẩm hoặc dịch vụ cho mọi người trên toàn thế giới, sử dụng dịch vụ giao hàng để vận chuyển sản phẩm đến khách hàng.

3.1.1 Một số nền tảng TMĐT

- **Amazon:**

Một trong những nền tảng lớn nhất thế giới, Amazon cho phép người bán và người mua thực hiện giao dịch sản phẩm từ hàng tiêu dùng, sách, đến đồ điện tử.

- **eBay:**

Hoạt động như một sàn giao dịch cho cả đấu giá và mua bán trực tiếp. eBay nổi bật với các sản phẩm độc đáo và dịch vụ toàn cầu.

- **Shopee (Đông Nam Á):**

Một nền tảng phổ biến tại khu vực Đông Nam Á, cung cấp các tính năng tiện lợi như trò chuyện trực tiếp giữa người mua và người bán.

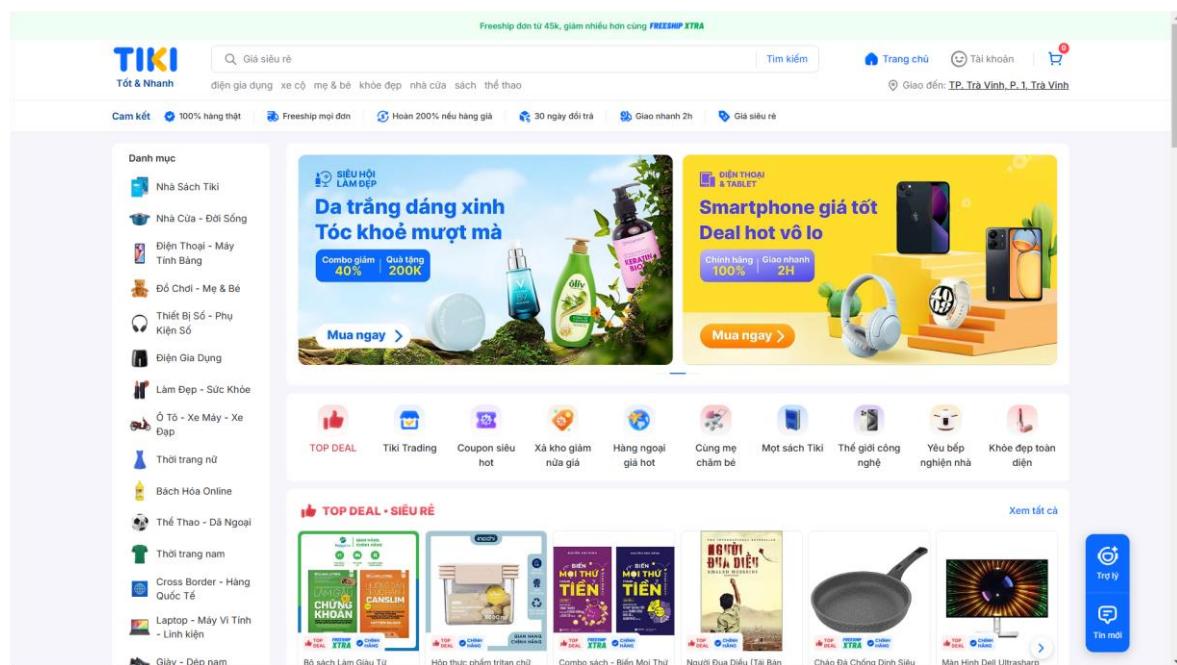
- **Lazada (Đông Nam Á):**

Thuộc tập đoàn Alibaba, Lazada tập trung mạnh vào thị trường Đông Nam Á với dịch vụ vận chuyển toàn diện.

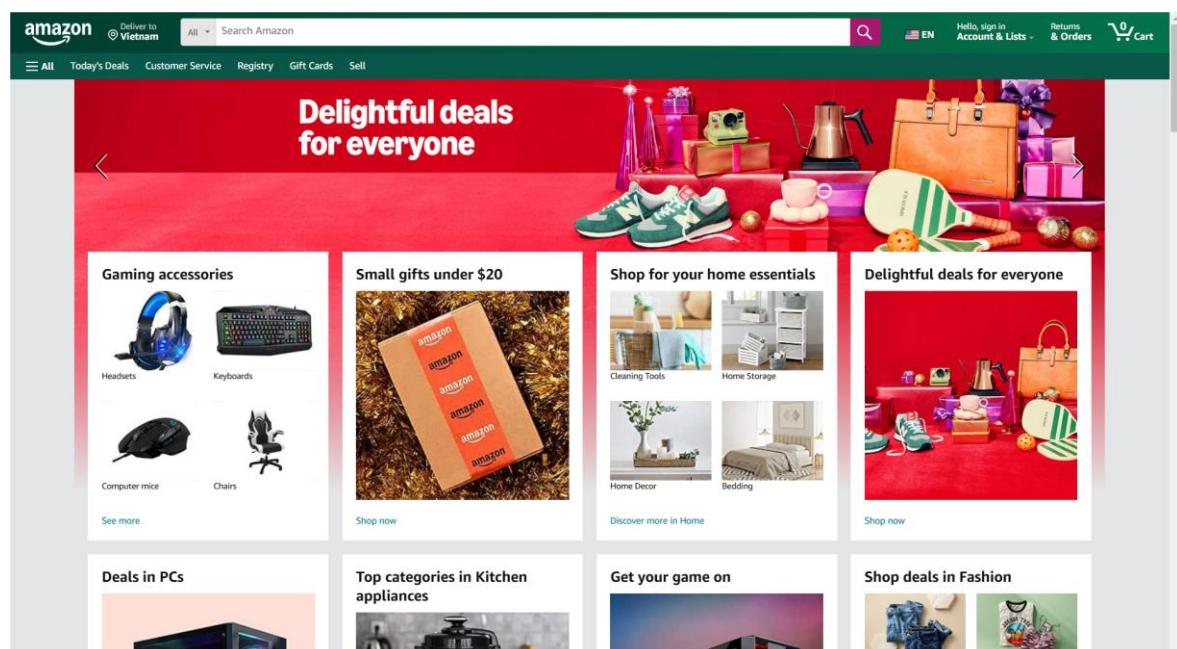
- **Tiki (Việt Nam):**

Là một nền tảng nổi bật tại Việt Nam, nổi tiếng với chất lượng hàng hóa và dịch vụ chăm sóc khách hàng tốt.

- Ví dụ về một số nền tảng



Hình 12 Nền tảng TMĐT Tiki



Hình 13 Nền tảng TMĐT Amazon

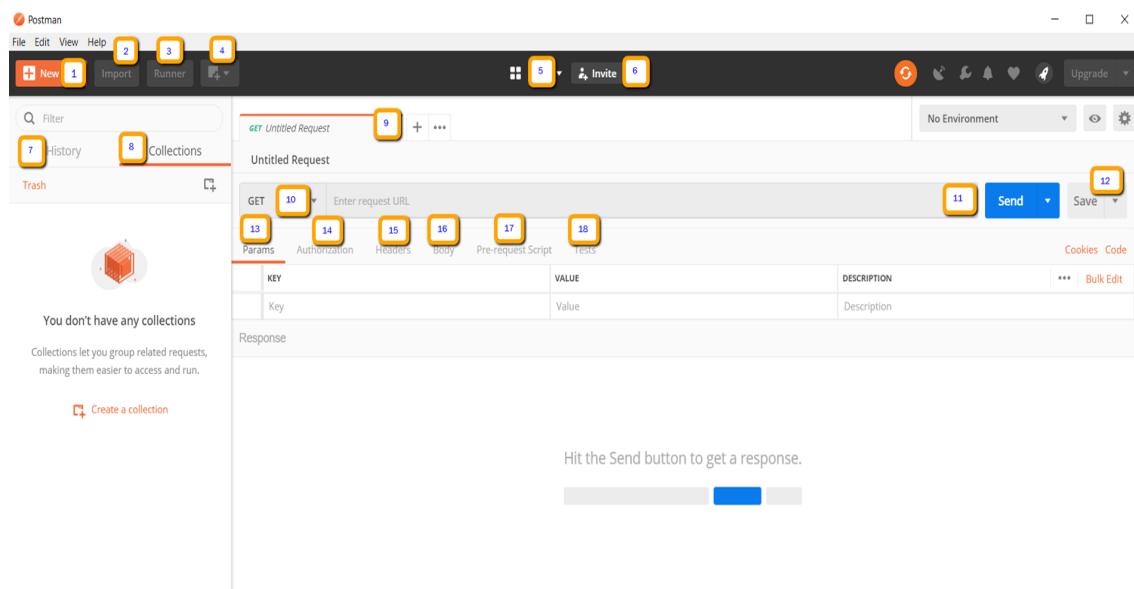
3.1.2 Tài liệu API của một số nền tảng TMĐT

- **Amazon:** Amazon Marketplace Web Service (MWS) API
 - **Chức năng:** truy xuất thông tin đơn hàng, danh sách sản phẩm, theo dõi tồn kho.
 - **Tài liệu API:** Amazon MWS Documentation
- **eBay:**
 - **Chức năng:** hỗ trợ tìm kiếm sản phẩm, quản lý danh sách bán hàng, theo dõi giao dịch.

- **Tài liệu API:** eBay Developer Program
- **Shopee (Đông Nam Á):**
 - **Chức năng:** cung cấp API để quản lý sản phẩm, đơn hàng, vận chuyển và báo cáo doanh thu.
 - **Tài liệu API:** Shopee Developers
- **Lazada (Đông Nam Á):**
 - **Chức năng:** hỗ trợ quản lý sản phẩm, đồng bộ dữ liệu tồn kho, xử lý đơn hàng, vận chuyển.
 - **Tài liệu API:** Lazada Developer Center
- **Tiki (Việt Nam):**
 - **Chức năng:** quản lý gian hàng, theo dõi đơn hàng, truy cập thông tin khách hàng (với quyền cho phép).
 - **Tài liệu API:** Tiki Seller Center

3.2 Sử dụng công cụ Postman kiểm thử API

3.2.1 Các thành phần trong Postman



Hình 14 Các thành phần trong Postman

1. **New** – tạo request, collection hoặc environment mới.
2. **Import** – import collection hoặc environment. Có các tùy chọn để import từ file, folder, link hoặc paste từ text thuần.
3. **Runner** – Kiểm tra tự động hóa có thể được thực hiện thông qua Runner cả collection.

4. **Open New** – Mở một tab mới, cửa sổ Postman hoặc cửa sổ Runner.
5. **My Workspace** – Tạo khu vực làm việc riêng hoặc cho một nhóm.
6. **Invite** – Cộng tác với nhiều thành viên bằng việc mời các thành viên.
7. **History** – Các request đã thực hiện mà bạn đã thực hiện sẽ được hiển thị trong History. Giúp bạn có thể lần theo các hành động bạn đã làm.
8. **Collections** – Tổ chức bộ thử nghiệm của bạn bằng cách tạo collection. Mỗi collection có thể có các thư mục con và nhiều yêu cầu. Request hoặc thư mục cũng có thể được trùng lặp.
9. **Tab Request** – Hiển thị tiêu đề của request mà bạn đang làm việc. Mặc định “Untitled Request” sẽ được hiển thị cho các request không có tiêu đề.
10. **HTTP Request** – Click vào đây sẽ hiển thị danh sách thả xuống với các request khác nhau như GET, POST, COPY, DELETE, v.v. Trong thử nghiệm, các yêu cầu được sử dụng phổ biến nhất là GET và POST.
11. **Request URL** – Còn được gọi là điểm cuối (endpoint), đây là nơi bạn sẽ xác định liên kết đến nơi API sẽ giao tiếp.
12. **Save** – Nếu có thay đổi đối với request, nhập vào Save là bắt buộc để những thay đổi mới sẽ không bị mất hoặc bị ghi đè.
13. **Params** – Đây là nơi bạn sẽ viết các tham số cần thiết cho một request, ví dụ như các cặp key – value.
14. **Authorization** – Để truy cập API, cần được cấp quyền. Nó có thể ở dạng tên người dùng và mật khẩu, bearer token, v.v.
15. **Headers** – Bạn có thể thiết lập các header như nội dung kiểu JSON tùy theo cách tổ chức của bạn.
16. **Body** – Đây là nơi chúng ta có thể tùy chỉnh chi tiết trong request thường được sử dụng trong request POST.
17. **Pre-request Script** – Đây là các tập lệnh sẽ được thực thi trước request. Thông thường, script tiền request (pre-request) cho cài đặt môi trường được sử dụng để đảm bảo các kiểm tra sẽ được chạy trong môi trường chính xác.

18. **Tests** – Đây là các script được thực thi khi request. Điều quan trọng là phải có các thử nghiệm như thiết lập các điểm checkpoint để kiểm tra trạng thái là ok, dữ liệu nhận được có như mong đợi không và các thử nghiệm khác.

3.2.2 Làm việc với GET request

The screenshot shows the Postman interface with the following details:

- Request Type:** GET (highlighted by red circle 1)
- URL:** https://jsonplaceholder.typicode.com/users (highlighted by red circle 2)
- Action:** Send (highlighted by red circle 3)
- Status Bar:** status: 200 OK, Time: 784 ms, Size: 6.27 KB (highlighted by red circle 4)
- Response Body (highlighted by red circle 5):**

```

1 {
2   "id": 1,
3   "name": "Leanne Graham",
4   "username": "Bret",
5   "email": "Sincere@april.biz",
6   "address": {
7     "street": "Kulas Light",
8     "suite": "Apt. 556",
9     "city": "Gwenborough",
10    "zipcode": "92998-3874",
11    "geo": {
12      "lat": "-37.3159",
13      "lng": "81.1496"
14    }
15  },
16  "phone": "1-770-736-8031 x56442",
17  "website": "hildegard.org",
18  "company": {
19    "name": "Romaguera-Crona",
20    "catchPhrase": "Multi-layered client-server neural-net",
21    "bs": "harness real-time e-markets"
22  },
23  {
24    "id": 2,
25    "name": "Ervin Howell",
26    "username": "Antonette",
27    "email": "Shanna@melissa.tv",
28    "address": {
29      "street": "Victor Plains",
30      "suite": "Suite 879",
31      "city": "Wisokyburgh",
32      "zipcode": "94194-1462",
33      "geo": {
34        "lat": "29.4043",
35        "lng": "-96.3412"
36      }
37    },
38    "phone": "010-654-5410",
39    "website": "anastasia.net",
40    "company": {
41      "name": "Deckow-Crist",
42      "catchPhrase": "Proactive didactic contingency",
43      "bs": "syndicated parallel architectures"
44    }
45  }

```

Hình 15 GET request

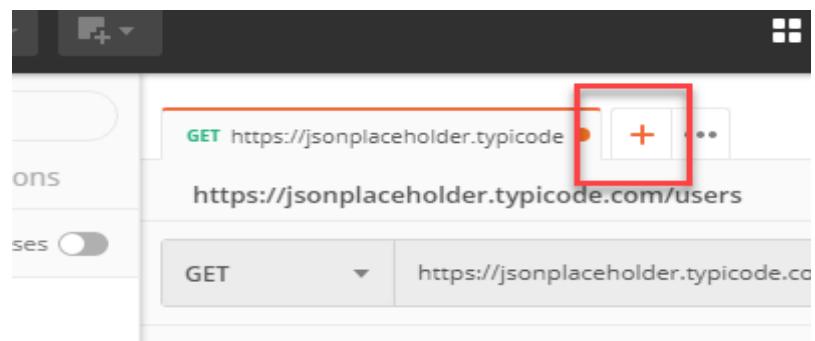
- Get request được sử dụng để lấy thông tin từ URL đã cho. Sẽ không có thay đổi được thực hiện đến cuối.

1. Đặt yêu cầu HTTP của bạn thành GET.
2. Trong trường URL yêu cầu nhập link
3. Nhập vào Gửi
4. Bạn sẽ thấy trạng thái 200 OK
5. Kết quả call API được trả về

3.2.3 Làm việc với POST request

- Post request khác với Get request vì có thao tác dữ liệu với người dùng thêm dữ liệu vào điểm cuối.

Bước 1: Click new tab để tạo request mới



Hình 16 Tạo request mới

Bước 2: Trong tab mới

1. Chuyển HTTP request thành POST.
2. Nhập liên kết url
3. Chuyển sang tab Body

Hình 17 Tab mới

1. Chuyển HTTP request thành POST.

2. Nhập liên kết url

3. Chuyển sang tab Body

Bước 3: Trong body

1. Click vào raw
2. Chọn JSON

Hình 18 Trong body

1. Click vào raw

2. Chọn JSON

Bước 4: Sao chép và dán một kết quả từ yêu cầu nhận được. Đảm bảo rằng mã đã được sao chép chính xác với dấu ngoặc vuông và dấu ngoặc nhọn. Có thể thay đổi các chi tiết khác như địa chỉ.

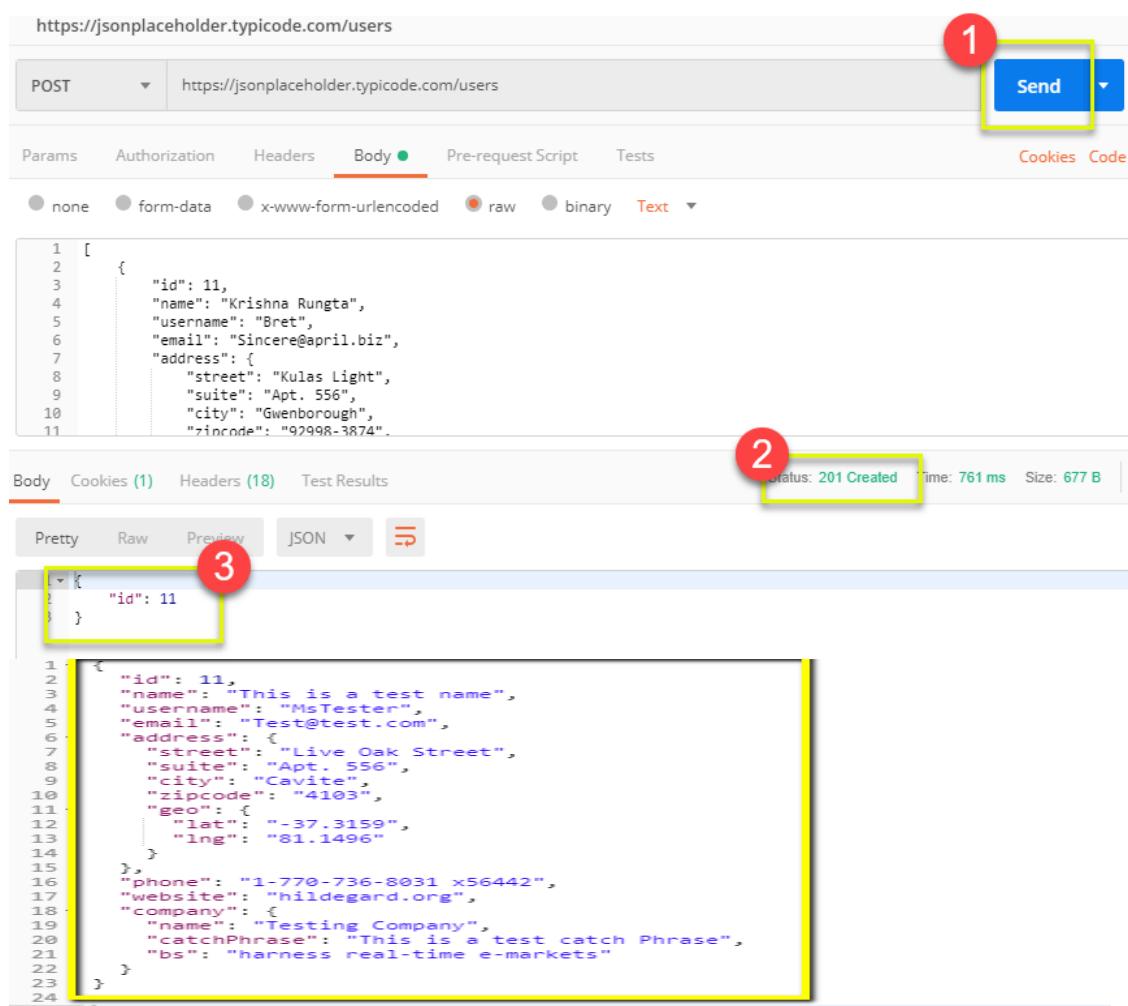
The screenshot shows the Postman application interface. At the top, there are two buttons: 'GET https://jsonplaceholder.typicode.com/users' (disabled) and 'POST https://jsonplaceholder.typicode.com/users'. Below the buttons is a URL input field with the value 'https://jsonplaceholder.typicode.com/users'. The main area is a POST request configuration. The 'Body' tab is selected, indicated by a red underline. Under the 'Body' tab, the 'Text' radio button is selected. A yellow box highlights the JSON data in the body:

```
[  
  {  
    "id": 11,  
    "name": "Krishna Rungta",  
    "username": "Bret",  
    "email": "Sincere@april.biz",  
    "address": {  
      "street": "Kulas Light",  
      "suite": "Apt. 556",  
      "city": "Gwenborough",  
      "zipcode": "92998-3874",  
      "geo": {  
        "lat": "-37.3159",  
        "lng": "81.1496"  
      }  
    },  
    "phone": "1-770-736-8031 x56442",  
    "website": "hildegard.org",  
    "company": {  
      "name": "Romaguera-Crona",  
      "catchPhrase": "Multi-layered client-server neural-net",  
      "bs": "harness real-time e-commerce"}
```

Hình 19 Sao chép và dán kết quả

- Post request phải có định dạng chính xác để đảm bảo dữ liệu được request sẽ được tạo. Đó là một cách thực hành tốt để sử dụng GET trước để kiểm tra định dạng JSON của yêu cầu.

Bước 5: Chạy kiểm tra



Hình 20 Kết quả trả về

1. Click Send
2. Trạng thái 201 Created được hiển thị
3. Dán dữ liệu vào body

3.3 Giải pháp thu thập dữ liệu API bằng Python

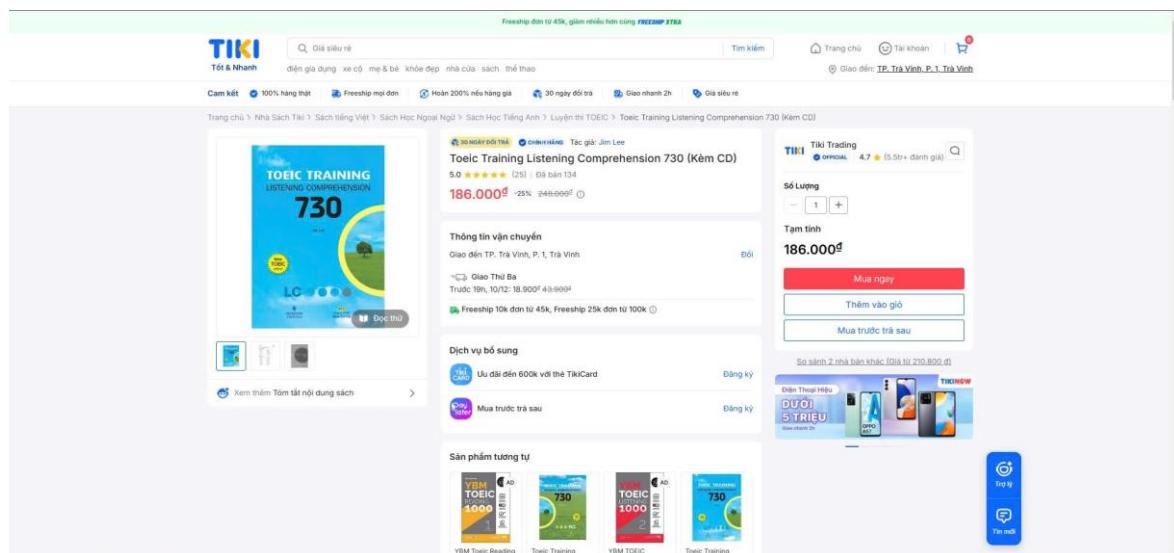
3.3.1 Lấy đường dẫn API và kiểm tra

a. Lựa chọn và xác định nền tảng TMĐT cần thu thập dữ liệu

Tùy vào mục đích và nhu cầu mà lựa chọn nền tảng TMĐT và cửa hàng, sản phẩm cần thu thập dữ liệu.

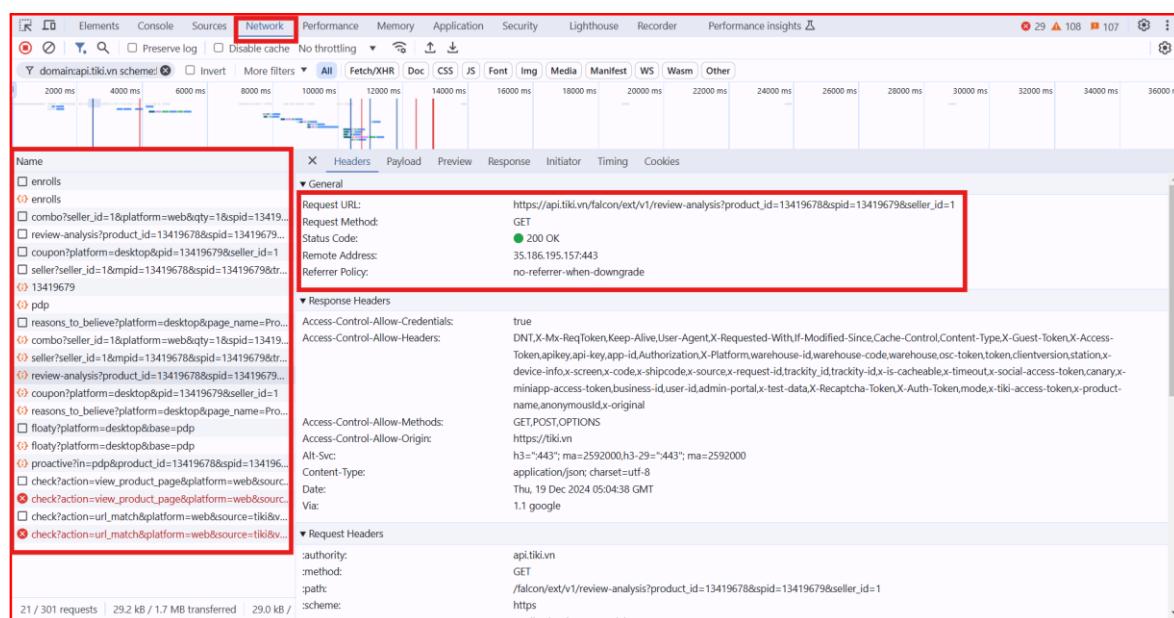
Ví dụ: Nền tảng TMĐT Tiki với một sản phẩm cụ thể

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử



Hình 21 Nền tảng TMĐT Tiki

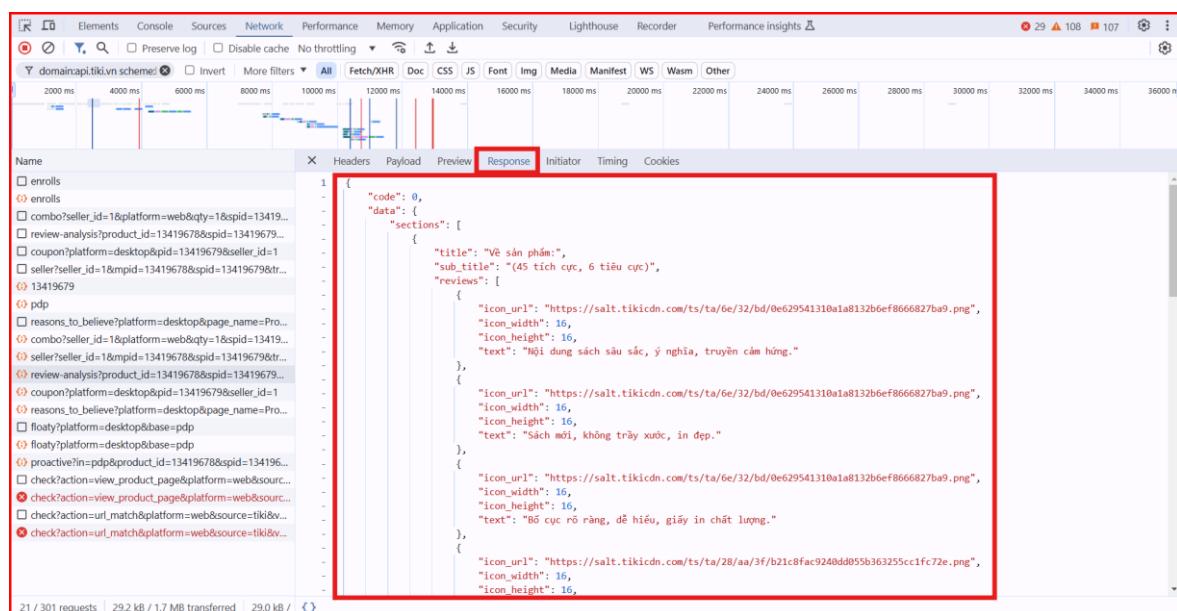
- b. Khi đã chọn được đối tượng cần thu thập ta kích chuột phải chọn Inspect (kiểm tra) hoặc ấn phím F12 để mở chức năng xem mã nguồn.
- c. Khi mở chức năng xem mã nguồn thì chọn Network sau đó kiểm tra có đường dẫn API hay không



Hình 22 Kiểm tra lấy đường dẫn API

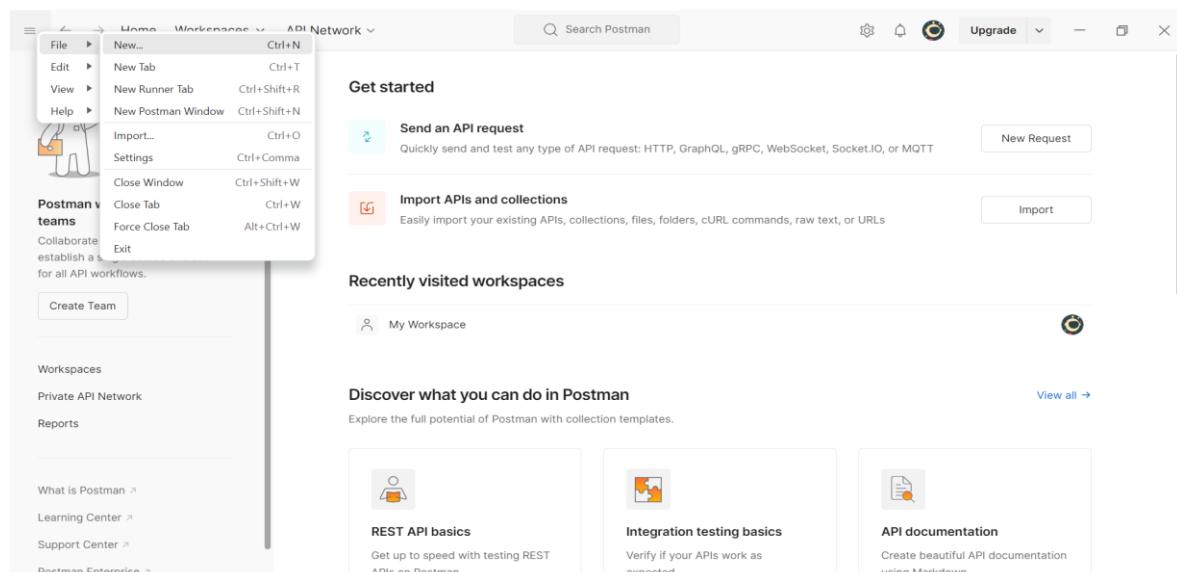
- d. Sau khi đã kiểm tra chọn Response để xem nội dung của đường dẫn

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử



Hình 23 Kiểm tra nội dung đường dẫn

e. Sử dụng phần mềm Postman để kiểm thử



Hình 24 Truy cập Postman

f. Dựa vào nội dung đã kiểm tra khi lấy đường dẫn sau đó điều chỉnh trong Postman để kiểm tra xem có thể thu thập được dữ liệu hay không

▼ General	
Request URL:	https://api.tiki.vn/falcon/ext/v1/review-analysis?product_id=13419678&spid=13419679&seller_id=1
Request Method:	GET
Status Code:	200 OK
Remote Address:	35.186.195.157:443
Referrer Policy:	no-referrer-when-downgrade

Hình 25 Nội dung kiểm tra

g. Sau khi đã điều chỉnh và kiểm tra thì sẽ có 2 trường hợp xuất hiện

Trường hợp 1: Đường dẫn không thể lấy được dữ liệu

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử

The screenshot shows the Postman interface with a GET request to <https://api.tiki.vn/integration/v2/categories?name=a&parent=320>. The request includes query parameters: Key, name (value: a), and parent (value: 320). The response status is 401 Unauthorized, with a response body containing an error message: "Missing tiki-api header or Authorization header".

Hình 26 Không lấy được dữ liệu

Trường hợp 2: Đường dẫn có thể lấy được dữ liệu

The screenshot shows the Postman interface with a GET request to https://api.tiki.vn/falcon/ext/v1/review-analysis?product_id=13419678&spid=13419679&seller_id=1. The request includes query parameters: product_id (value: 13419678), spid (value: 13419679), and seller_id (value: 1). The response status is 200 OK, with a JSON response body containing review analysis data.

Hình 27 Lấy được dữ liệu

3.3.2 Sử dụng Python để thu thập dữ liệu

a. Có thể sử dụng Visual Studio hoặc Google Colab

The screenshot shows the Google Colaboratory homepage. It features a large image of a modern building at night, a central banner with the text "Google Colaboratory" and a brief description, and two side panels: "News and Guidance" and "Explore Notebooks".

Hình 28 Sử dụng Google Colab

b. Lập trình lấy dữ liệu

```
import requests
import json
import matplotlib.pyplot as plt

# URL API và thông tin headers
url = "https://api.tiki.vn/falcon/ext/v1/review-analysis?product_id=715598&spid=717028&seller_id=1"
headers = {
    "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36"
}

try:
    # Gửi yêu cầu GET đến API
    response = requests.get(url, headers=headers)

    # Kiểm tra phản hồi từ server
    if response.status_code == 200:
        try:
            # Lấy dữ liệu JSON từ phản hồi
            data = response.json()

            # Lưu toàn bộ dữ liệu vào file JSON
            output_file = "full_data.json"
            with open(output_file, "w", encoding="utf-8") as file:
                json.dump(data, file, ensure_ascii=False, indent=4)

            print(f"Toàn bộ dữ liệu đã được lưu vào file: {output_file}")

            # In toàn bộ dữ liệu ra màn hình
            print("Toàn bộ dữ liệu phản hồi từ API:")
            print(json.dumps(data, ensure_ascii=False, indent=4))

            # Kiểm tra và xử lý dữ liệu rating_distribution
            rating_distribution = data.get("rating_distribution")
            service_rating_distribution = data.get("service_rating_distribution")

        except json.JSONDecodeError as e:
            print("Lỗi khi phân tích dữ liệu JSON:", e)
    else:
        # In lỗi nếu phản hồi không thành công
        print(f"Yêu cầu không thành công. Mã lỗi: {response.status_code}")
        print(f"Nội dung phản hồi: {response.text}")

except requests.exceptions.RequestException as e:
    print(f'Lỗi khi gửi yêu cầu đến API: {e}')
```

Hình 29 Lập trình lấy dữ liệu API

c. Chức năng và ý nghĩa của từng phần

- Nhập các thư viện cần thiết:

- **requests**: thư viện hỗ trợ gửi các yêu cầu HTTP, rất phổ biến trong việc làm việc với API.
- **json**: thư viện để xử lý dữ liệu JSON (chuyển đổi từ JSON sang Python object hoặc ngược lại).
- **matplotlib.pyplot**: thư viện vẽ đồ thị, tuy nhiên, trong đoạn mã hiện tại, chưa có phần sử dụng plt.

- Khai báo URL API thông tin headers:

- **url**: địa chỉ API để lấy dữ liệu, với các tham số sản phẩm như:
product_id=715598: ID của sản phẩm.

spid=717028: Một dạng mã phụ của sản phẩm.

seller_id=1: ID của người bán.

- **headers:** bao gồm thông tin User-Agent mô phỏng trình duyệt để tránh bị API chặn khi gửi yêu cầu từ script.

- Thực hiện yêu cầu HTTP GET đến API:

- Gửi yêu cầu GET đến API với URL và headers đã định nghĩa.

- Kiểm tra trạng thái phản hồi:

- **Kiểm tra mã trạng thái:** HTTP **200** nghĩa là yêu cầu thành công. Nếu không phải **200**, mã sẽ in thông báo lỗi và nội dung phản hồi từ server.

- Xử lý dữ liệu JSON từ phản hồi:

- Chuyển đổi nội dung phản hồi (dạng chuỗi JSON) thành object Python (**dict**).

- Lưu dữ liệu vào file JSON:

- Ghi toàn bộ dữ liệu JSON vào file **full_data.json**:

ensure_ascii=False: Đảm bảo các ký tự Unicode (chẳng hạn tiếng Việt) không bị chuyển đổi sang dạng mã ASCII.

indent=4: Định dạng dữ liệu với thụt lè 4 ký tự để dễ đọc.

- In dữ liệu JSON ra màn hình:

- **json.dumps:** chuyển đổi object Python (như dict) thành chuỗi JSON và in ra màn hình với định dạng dễ đọc.

- Xử lý trường dữ liệu:

- **data.get():** truy xuất các trường từ dữ liệu JSON trả về, tránh lỗi nếu trường không tồn tại.

- **Mục đích:**

rating_distribution: Có thể chứa thông tin phân bố đánh giá sản phẩm theo từng mức sao (1-5).

service_rating_distribution: Có thể liên quan đến đánh giá dịch vụ từ nhà bán hàng.

- Xử lý lỗi JSON Decode:

- Xử lý trường hợp phản hồi không phải dạng JSON hợp lệ

- Xử lý lỗi yêu cầu HTTP:

- Bắt lỗi chung khi xảy ra vấn đề trong quá trình gửi yêu cầu đến API, như: Kết nối thất bại, Server không phản hồi, lỗi mạng.

d. Kết quả

```
Toàn bộ dữ liệu phản hồi từ API:
{
    "code": 0,
    "data": {
        "sections": [
            {
                "title": "Về sản phẩm:",
                "sub_title": "(11 tích cực, 0 tiêu cực)",
                "reviews": [
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/6e/32/bd/0e629541310a1a8132b6ef8666827ba9.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Sách được đánh giá là hay, phù hợp để luyện nghe Toeic và có nội dung tương đối phù hợp với đề thi thật."
                    },
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/6e/32/bd/0e629541310a1a8132b6ef8666827ba9.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Sách có sách scripts và đáp án riêng, thiết kế đẹp và đầy đủ ebook và bản script và đĩa."
                    },
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/6e/32/bd/0e629541310a1a8132b6ef8666827ba9.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Một số người mua hài lòng với việc sách mới, giao hàng nhanh và có giá sale hấp dẫn."
                    }
                ],
                "title": "Về dịch vụ:",
                "sub_title": "(4 tích cực, 2 tiêu cực)",
                "reviews": [
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/6e/32/bd/0e629541310a1a8132b6ef8666827ba9.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Đa số khách hàng hài lòng với dịch vụ giao hàng nhanh."
                    },
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/6e/32/bd/0e629541310a1a8132b6ef8666827ba9.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Một số người mua đánh giá tích cực về dịch vụ hỗ trợ sau bán hàng."
                    },
                    {
                        "icon_url": "https://salt.tikicdn.com/ts/ta/28/aa/3f/b21c8fac9240dd055b363255cc1fc72e.png",
                        "icon_width": 16,
                        "icon_height": 16,
                        "text": "Tuy nhiên, có một số người mua không có đầu đọc CD và yêu cầu file audio, nhưng không nhận được phản hồi."
                    }
                ]
            }
        ]
    }
}
```

Hình 30 Kết quả trả về

0 19946648 Cực kì hài lòng Cùng 2 năm rồi mình mới lại mua sách lật và sau mỗi trán phản nản suốt mấy năm liền thì tiki đã cố gắng cải thiện khâu đóng gói rồi. Lần này mình đặt 2 cuốn được bọc nhiều lớp ni *** chống nước*** và đựng rất kín đáo, chắc chắn. Mới đọc xong mà vẫn còn nguyên vẹn. Franklin cũng dày dặn ai mà cũng vừa vui vừa vầy.			
1 8838997 Cực kì hài lòng Benjamin Franklin thì bạn bè quá quen, ông là một trong những người được ngưỡng mộ và nhắc đến khá nhiều trên thế giới. Sách được bọc kỹ và mới. Giấy như SGK :))			
2 10446593 Cực kì hài lòng Mạch tư duy: Từ nguồn cội đến lúc ông trưởng thành và trải qua những biến cố cùng như bài học trong cuộc đời. Điểm tiêu biểu: Cách ông tự quản chiếu cuộc đời mình và đi theo những nguyên tắc do ông tự đề ra (được học khi đọc sách nữa).			
3 18985821 Cực kì hài lòng Nhìn xét cả nhân: Cách ông áp dụng khá giống với trong các đạo nhắc đến sự quản chiếu bản thân. Với những đóng góp to lớn cho nhân loại bằng cách luôn nghĩ cách giúp tốt hơn cho con người, minh nghĩa Love!			
4 11250182 Cực kì hài lòng Một cuốn sách hay và một ví nhân lịch sử, là một người yêu thích những cuốn sách về vĩ nhân, minh cảm cách già Nguyễn Thúy Khanh Chưng và NXB Lu Động đưa đến độc giả những cuốn sách hay và có giá			
5 13218994 Cực kì hài lòng Mình mua với mong muốn đọc cuốn sách rất tuyệt vời nhất. Ngoài sự vĩ đại của Benjamin, cuốn sách còn giúp minh hiểu hơn về cuộc sống. Hiểu rõ vì sao có người thành công và vì sao có người thất bại, sự			
6 6442470 Cực kì hài lòng Một cuốn tự truyện rất đời thường với những chuyện rất gần của Benjamin Franklin.			
7 20002617 Cực kì hài lòng Trong cuốn sách chúng ta sẽ thấy và có thể hỏi được rất nhiều điều ở ông với các lập luận về chính trị, đạo đức, giáo dục, tôn giáo, kinh doanh... trong hành trình của một cậu thiếu niên trẻ.			
8 19702894 Cực kì hài lòng Sách khá đep nhung thực sự báu diệu cỏ chưa được chau chuốt vè ngôn từ nên đọc nhiều đoạn long cung và cỗ thiêu từ.Hi vọng bản tái bản mới tốt hơn.			
9 12846816 Cực kì hài lòng Mình vừa nhận sách, vẫn nguyên màng bọc nilon, sách đep, chờ in rõ ràng, chất giấy tốt.			

Hình 31 Kết quả được sắp xếp ở dạng bảng

3.4 Giải pháp thu thập dữ liệu thông qua phần mềm quản lý đám mây

3.4.1 Một số phần mềm quản lý đám mây hỗ trợ thu thập API

- Google Cloud Platform (GCP)

- Ứng dụng: kết hợp với Cloud Functions hoặc BigQuery để thu thập và xử lý dữ liệu từ API.
- Tính năng:

Lưu trữ dữ liệu trong Google Storage.

Tích hợp trực tiếp với các dịch vụ phân tích dữ liệu lớn.

- AWS (Amazon Web Services)

- Ứng dụng: sử dụng AWS Lambda để gọi API định kỳ và lưu dữ liệu trong S3.

- Tính năng:

Quản lý dữ liệu phi cấu trúc.

Kết hợp với các công cụ như Redshift hoặc Athena để phân tích dữ liệu.

- Microsoft Azure

- Ứng dụng: Azure Logic Apps hoặc Azure Functions để tự động hóa thu thập API.

- Tính năng:

Hỗ trợ kết nối với API REST hoặc SOAP.

Xử lý dữ liệu thời gian thực.

- Zapier

- Ứng dụng: công cụ không cần lập trình (No-code) hỗ trợ gọi API và lưu dữ liệu lên Google Sheets, CRM hoặc phần mềm khác.

- Tính năng:

Hỗ trợ hàng trăm tích hợp API với các nền tảng thương mại điện tử.

Dễ sử dụng và cấu hình.

- Integromat (Make)

- Ứng dụng: một nền tảng tự động hóa mạnh mẽ với các kịch bản phức tạp để gọi API và xử lý dữ liệu.

- Tính năng:

Hỗ trợ tích hợp API tùy chỉnh.

Dễ dàng chuyển dữ liệu giữa các hệ thống khác nhau.

- Acho

- Ứng dụng: phù hợp với nhiều lĩnh vực khác nhau nhờ khả năng tích hợp, xử lý, và phân tích dữ liệu linh hoạt.

- Tính năng:

Hỗ trợ bảo mật cao, tự động hóa quy trình và phân tích dữ liệu thời gian thực.

Tích hợp đa nguồn, xử lý dữ liệu tự động và trực quan hóa qua dashboard.

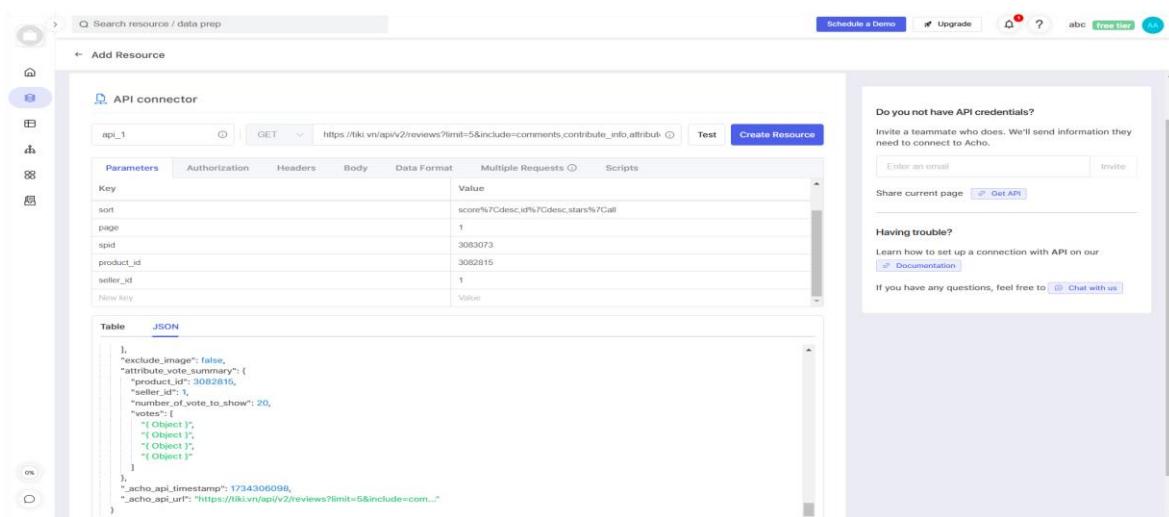
3.4.2 Quy trình thực hiện

a. Cấu hình API endpoint (điểm cuối)

- API endpoint có thể phức tạp. Nó đòi hỏi sự hiểu biết tốt về thiết kế kỹ thuật của nó để phân tích nó thành một hệ thống một cách hiệu quả.

Ví dụ: https://api.tiki.vn/falcon/ext/v1/review-analysis?product_id=715598&spid=717028&seller_id=1

- Có thể trực tiếp dán URL endpoint vào hộp nhập URL. Hoặc bạn có thể nhập các tham số tương ứng. Sau đó, bạn có thể nhấp vào "Test" để xem yêu cầu có thành công không. Nếu tất cả các cấu hình đều chính xác, bạn sẽ thấy kết quả như hiển thị bên dưới.



Hình 32 Cấu hình API

- Nếu không thấy bất kỳ kết quả nào từ API của mình hoặc cảnh báo hệ thống hiện ra, nên kiểm tra lại tài liệu API và đảm bảo rằng tất cả các cấu hình đã được điền đầy đủ.

- Một số sự cố phổ biến khi kết nối API:

API yêu cầu một quyền hạn cụ thể, chẳng hạn như: Quyền hạn cơ bản hoặc mã thông báo người mang. → Chỉ định quyền hạn này trong tab “Authorization”

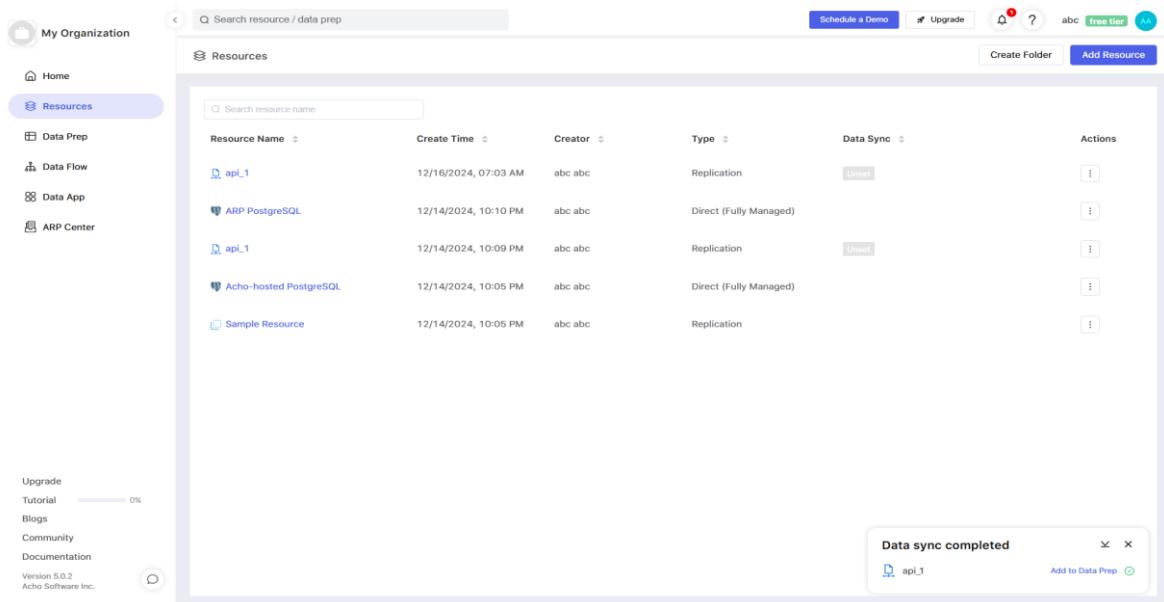
API cần điền một số thông tin trong tiêu đề. → Chỉ định quyền hạn này trong tab “Headers”

Kết quả của API là tệp CSV có thể tải xuống. → Di chuyển tab “Data Format” và chọn “CSV”

API chứa dấu ngoặc nhọn {}. → Giá trị trong dấu ngoặc là động. Cần phải cung cấp danh sách các giá trị cho hệ thống để hệ thống có thể thay đổi giá trị bên trong dấu ngoặc. Chúng ta sẽ nói về điều này sau trong bài viết này.

b. Tạo một tài nguyên API.

- Sau khi xác nhận kết quả API, nhấp vào “Create Resource” để xây dựng kết nối API. Tài nguyên API sẽ hiển thị trên trang Resources (tài nguyên).

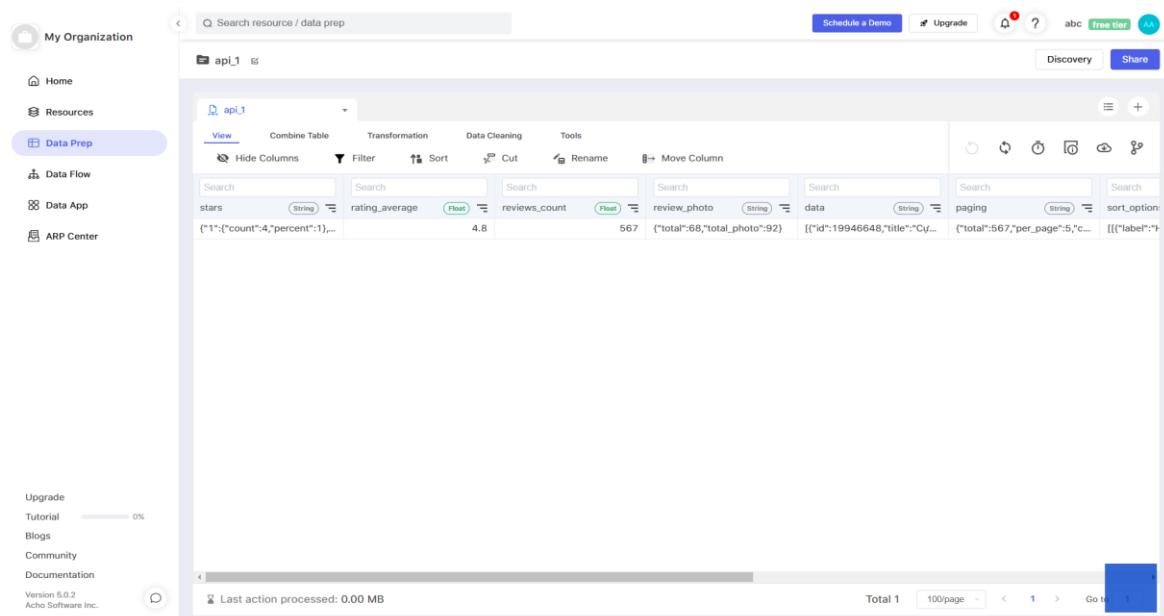


Hình 33 Tạo tài nguyên API

Lưu ý: Nếu muốn API Connector tự động cập nhật, có thể vào tab “Data Sync” trong phần cài đặt và bật Scheduler. Chế độ cập nhật cho phép bạn xác định cách cập nhật bảng: thêm dữ liệu mới vào dữ liệu hiện có (Incremental) hoặc thay thế dữ liệu hiện có bằng dữ liệu mới (Full Refresh).

c. Lưu trữ dữ liệu vào cơ sở dữ liệu

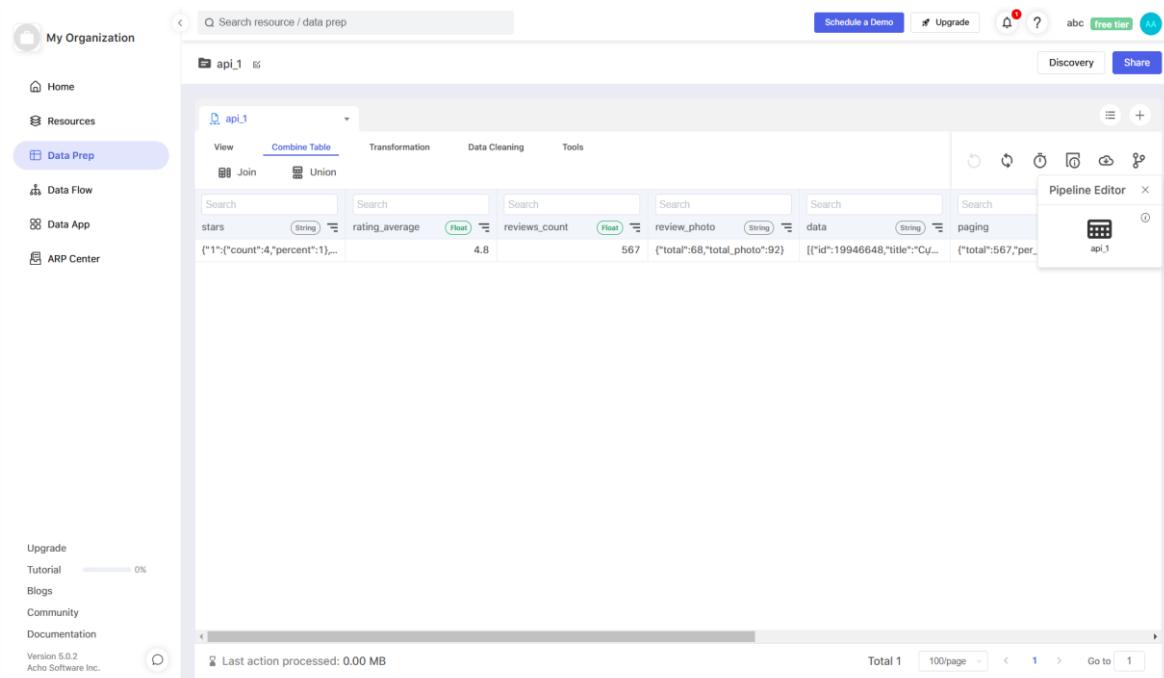
- Sau khi dữ liệu được kéo từ API và quá trình đồng bộ ban đầu hoàn tất, dữ liệu sẽ được lưu trữ trong cơ sở dữ liệu dạng bảng sẽ đồng bộ với tài nguyên. Khi API cập nhật, tài nguyên sẽ cập nhật nội dung của nó và bảng dự án cũng sẽ phản ánh bản cập nhật đó.



Hình 34 Lưu trữ API trong cơ sở dữ liệu

d. Chuyển đổi dữ liệu API

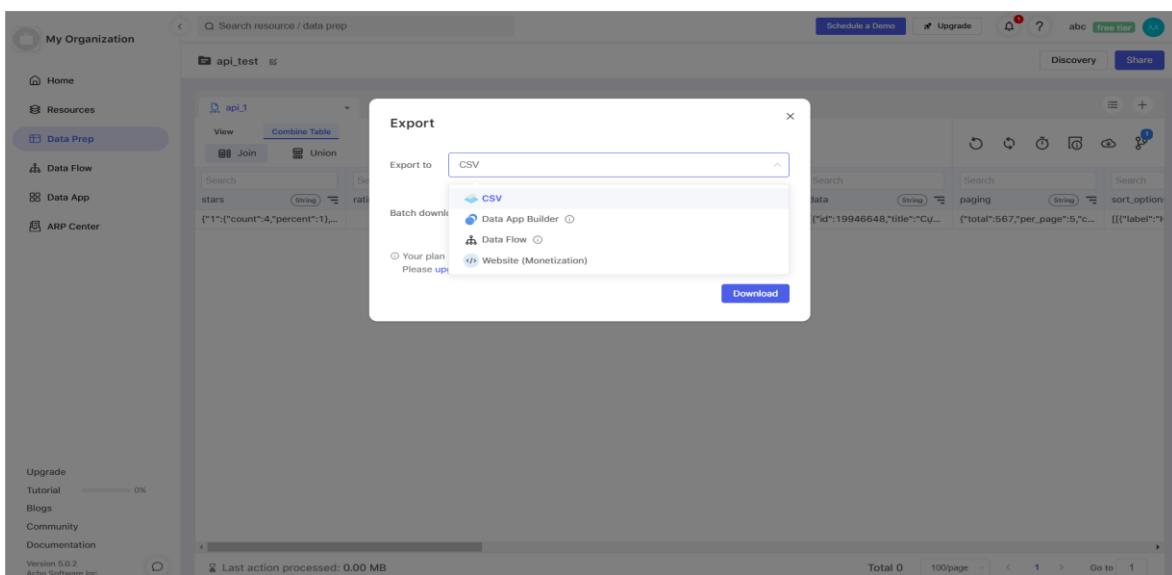
- Trong cơ sở dữ liệu, có thể tự do chuyển đổi nó thành định dạng bảng mong muốn. Dữ liệu API được lưu trữ theo định dạng dữ liệu lồng nhau, chẳng hạn như JSON hoặc mảng. Có thể sử dụng hành động "Flatten" để làm phẳng dữ liệu thành định dạng bảng. Ngoài ra, có một số hành động dựa trên SQL tích hợp có thể áp dụng cho dữ liệu được kéo từ API.



Hình 35 Chuyển đổi dữ liệu

e. Xuất dữ liệu

- Lựa chọn dữ liệu phù hợp và chọn Download để tải dữ liệu về máy.



Hình 36 Xuất dữ liệu

3.5 Giải pháp thu thập dữ liệu thông qua việc bán hàng

- Đăng ký hoặc đăng nhập tài khoản nếu có.

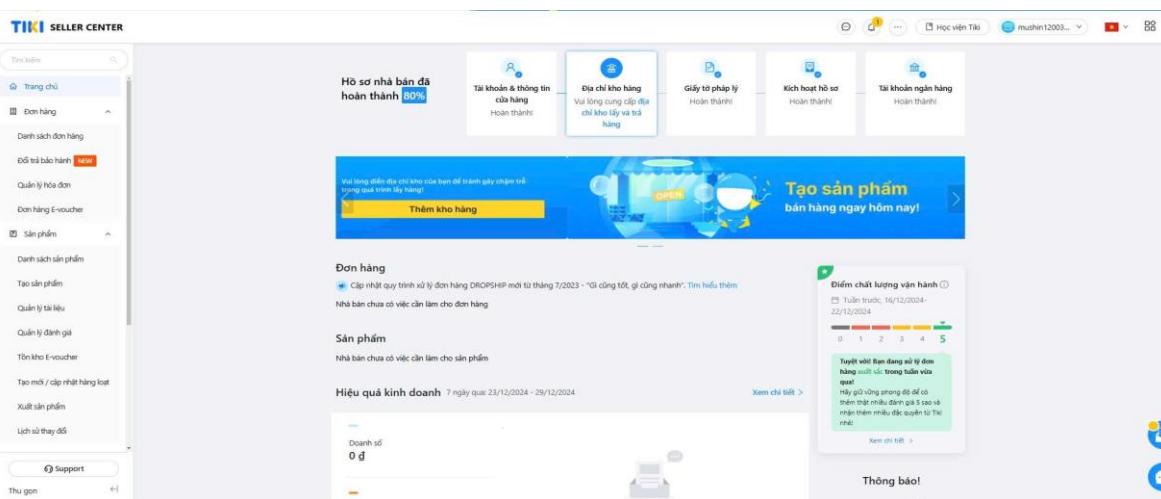


Hình 37 Đăng ký hoặc đăng nhập tài khoản bán hàng

Hình 38 Điện thông tin đăng ký

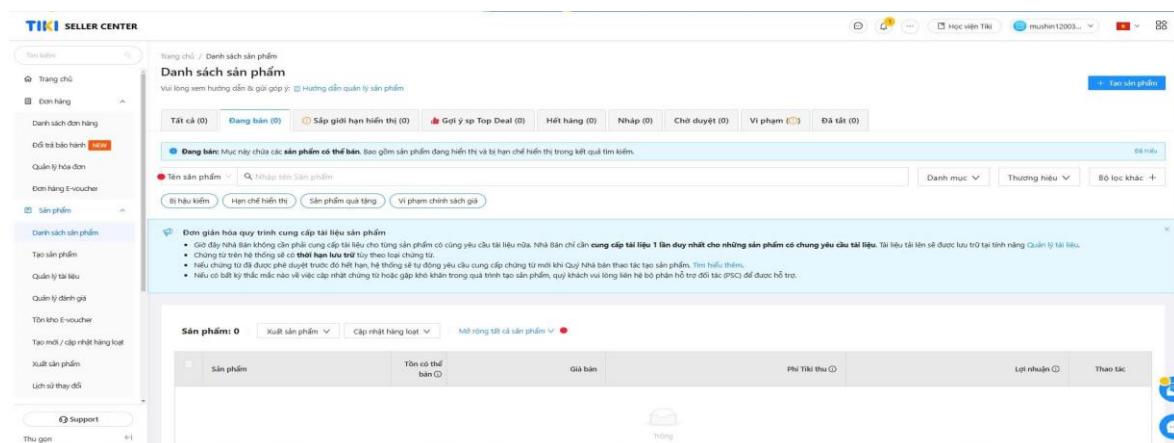
- Sau khi đăng ký và đăng nhập vào tài khoản thì điền các thông tin được yêu cầu cập nhật.

Tìm hiểu giải pháp thu thập dữ liệu thông qua API của các nền tảng thương mại điện tử

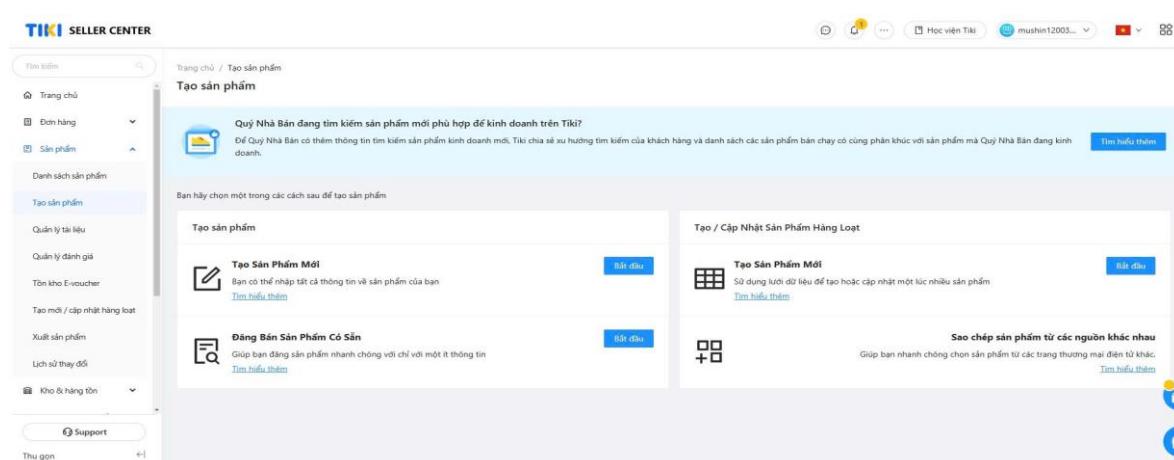


Hình 39 Trang chủ

- Sau khi cập nhật đầy đủ thông tin được yêu cầu chọn và danh mục Sản phẩm sao đó chọn vào Danh sách sản phẩm click vào nút Tạo sản phẩm để tạo sản phẩm bán hàng.

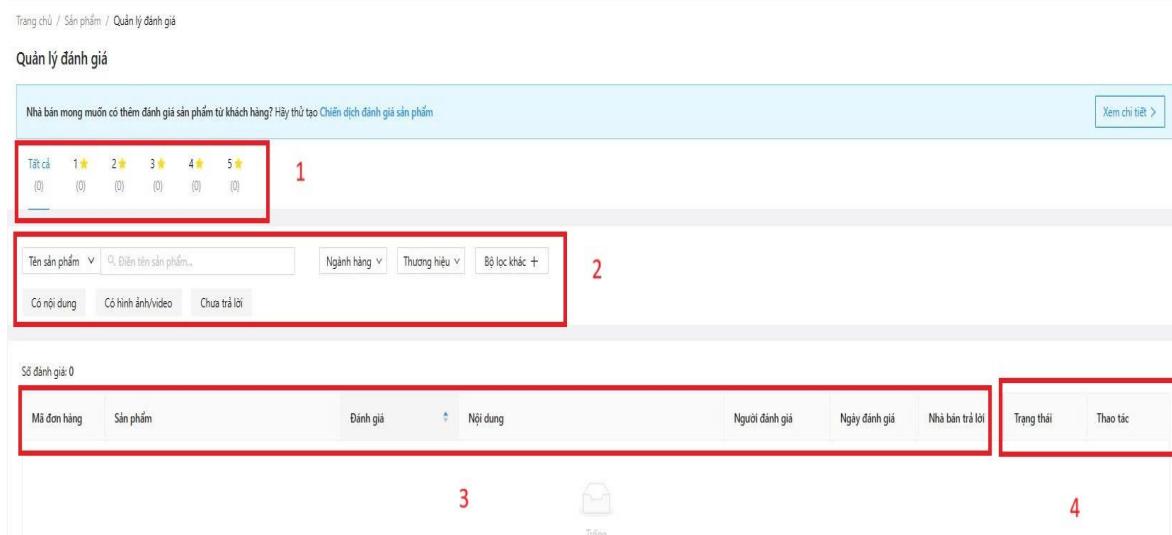


Hình 40 Danh sách sản phẩm



Hình 41 Tạo sản phẩm

- Sau khi sản phẩm được bán và được đánh giá chọn vào mục Sản phẩm sau đó chọn Quản lý đánh giá



Hình 42 Quản lý đánh giá

1. Xem phân loại đánh giá theo sao.
2. Lọc các đánh giá.
3. Đánh giá có ngày đánh giá gần nhất sẽ được hiển thị đầu tiên. Có thể sắp xếp thứ tự đánh giá theo số sao (1 sao – 5 sao) hoặc theo ngày đánh giá.
4. Có thể chọn “trả lời” để phản hồi đánh giá của khách hàng ở cột **Thao tác**. Hoặc chọn “báo cáo” khi nhận được đánh giá không phù hợp ở cột **Trạng thái**.
 - Sản phẩm được bán và được đánh giá từ khách hàng ta có thể thu được dữ liệu API bán hàng từ những sản phẩm được mua và được khách hàng đánh giá.

3.6 Giải pháp thu thập dữ liệu thông qua việc hợp tác với các cửa hàng

3.6.1 Đăng ký ứng dụng khách hàng

Ứng dụng khách hàng của bạn cần phải được đăng ký trước khi có thể sử dụng. Nền tảng nhận dạng Tiki Marketplace chỉ thực hiện quản lý nhận dạng và quyền truy cập cho các ứng dụng đã đăng ký. Do đó, bước đầu tiên là đăng ký ứng dụng. Ứng dụng đóng vai trò quan trọng trong Tiki Marketplace và có 2 loại ứng dụng

- Ứng dụng công cộng
- Ứng dụng nội bộ

Loại ứng dụng sẽ quyết định cách ứng dụng khách hàng của bạn tương tác với Tiki Marketplace và bạn không thể thay đổi loại ứng dụng sau khi tạo, vì vậy hãy đảm bảo rằng bạn hiểu rõ sự khác biệt và ứng dụng bạn cần chính xác là gì?

Để đăng ký ứng dụng, trước tiên, bạn hãy vào bảng điều khiển Tiki OpenAPI và điều hướng đến mục Application Create

(<https://open.tiki.vn/web/console/apps/create>)

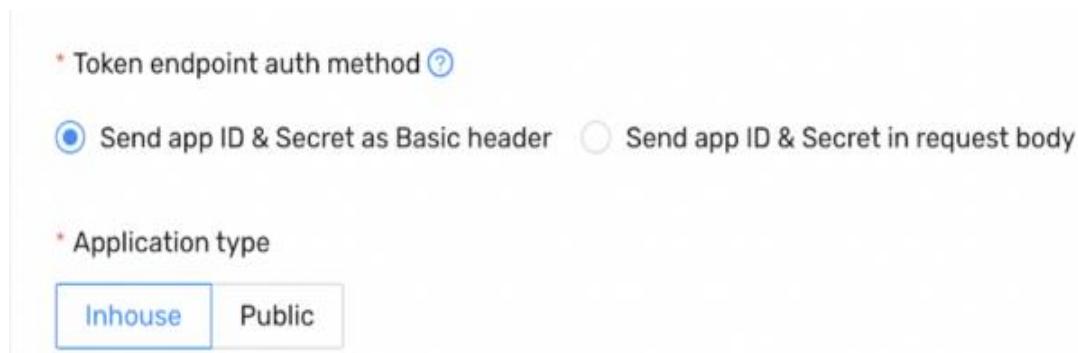
Sau đó, điền thông tin về ứng dụng mới tạo. Thông tin này bao gồm:

- Tên ứng dụng – Tên hiển thị cho ứng dụng của bạn. Nên sử dụng tên ngắn gọn, độc đáo và hấp dẫn.
- Tóm tắt – Tóm tắt ngắn gọn về những gì ứng dụng sẽ thực hiện.
- Mô tả – Mô tả chi tiết về chức năng của ứng dụng.
- URL trang chủ của ứng dụng – URL có thể truy cập vào trang chủ của ứng dụng.
- Biểu tượng ứng dụng – Hình ảnh PNG chất lượng (tệp PNG, kích thước 512x512)

The screenshot shows the 'Create Application' page of the Tiki Developer Platform. The left sidebar has a dark theme with blue highlights for 'Console', 'Applications' (which is expanded to show 'Application List' and 'Application Create'), and 'API Explorer'. The main content area has a light background. It contains fields for 'Application name', 'Summary', 'Description', 'Application homepage URL', 'Application icon' (with upload and file type restrictions), 'Token endpoint auth method' (radio buttons for 'Send app ID & Secret as Basic header' and 'Send app ID & Secret in request body'), and 'Application type' (radio buttons for 'Inhouse' and 'Public'). The 'Application name' field is pre-filled with 'Application name seen by end users'. The 'Summary' and 'Description' fields are empty. The 'Application homepage URL' field contains 'http://example.com/'. The 'Application icon' section shows a placeholder for an uploaded file. The 'Token endpoint auth method' section shows the first radio button is selected. The 'Application type' section shows the first radio button is selected.

Hình 43 Trang tạo ứng dụng

- Sau khi điền đầy đủ thông tin thì phải chọn phương thức xác thực điểm cuối Token và loại Ứng dụng.



Hình 44 Chọn phương thức

Phương pháp xác thực điểm cuối mã thông báo để chỉ định cách xác thực máy khách OAuth 2.0 tại điểm cuối mã thông báo. Để tìm hiểu thêm về phương pháp xác thực điểm cuối mã thông báo.

Với phương pháp xác thực điểm cuối token, bạn có thể chuyển đổi giữa “basic header” và “request body”. Với loại ứng dụng, bạn có thể nhấp vào nút “Inhouse” hoặc “Public”.

Không thể thay đổi phương thức xác thực điểm cuối mã thông báo hoặc loại ứng dụng sau khi ứng dụng được tạo.

- Sau khi đã chọn loại đơn đăng ký, cần phải cung cấp một số thông tin bổ sung cần thiết cho từng loại đơn đăng ký.

The screenshot shows a configuration form for a public application. It includes fields for "Application type" (set to "Public"), "Application scopes" (with options like "Create & manage multichannel orders", "Manage normal deals of a seller", and "Manage seller warehouses"), "Policy URL" (set to "https://yourappdomain.com/policy"), "Term of Services URL" (set to "https://yourappdomain.com/tos"), and "Redirect URIs" (containing "https://yourappdomain.com/tiki/callback", "https://yourappdomain.com/users/callback", and "http://localhost:8000/tiki/callback"). A note at the bottom states "These will be valid URLs for Oauth callback." At the bottom are "Submit" and "Reset" buttons.

Hình 45 Ứng dụng công cộng

* Application type
 Inhouse Public

An in-house app is connected to exactly one store. This app needs seller to confirm the connection before it can access that seller's store.

* Seller ID
1122334455

* Seller name
Your Shop Name

Hình 46 Ứng dụng nội bộ

Kết nối và phê duyệt

- Ứng dụng công cộng

Ứng dụng phải được người kiểm duyệt của Tiki xem xét để xác minh và có thể phải đợi 2–24 giờ vì ứng dụng là công khai đối với tất cả người bán cũng là người dùng Tiki, vì vậy cần xem xét một số thông tin bắt buộc. Bao gồm quyền riêng tư, điều khoản dịch vụ và các thông tin khác có liên quan đến ứng dụng.

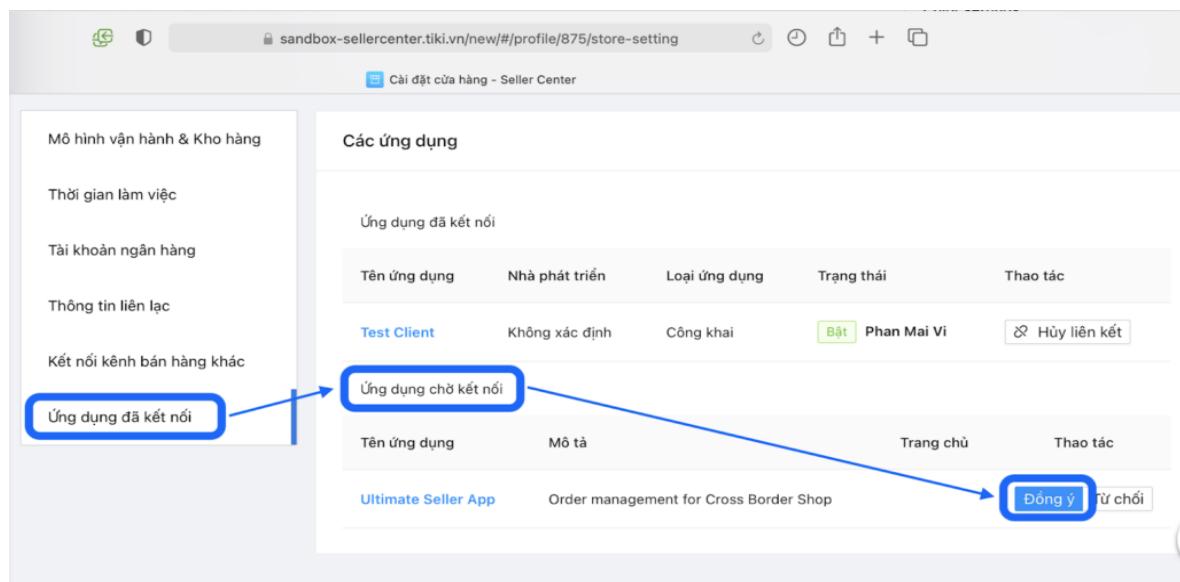
Trong trường hợp này, đơn đăng ký đã bị từ chối. Bạn có thể vào [bảng điều khiển OpenAPI](#), điều hướng đến **Application List => Detail** và chỉnh sửa thông tin đơn đăng ký của bạn và yêu cầu xác minh lại.

- Ứng dụng nội bộ

Ứng dụng được kết nối với các cửa hàng của người bán để đảm bảo rằng người bán cho phép ứng dụng của khách hàng truy cập vào dữ liệu của cửa hàng của họ. Vì vậy, không cần phải được người kiểm duyệt của Tiki xác minh nhưng cần yêu cầu người bán xác nhận ứng dụng và hướng dẫn người bán xác nhận và cho phép ứng dụng kết nối với cửa hàng của họ bằng cách làm theo hướng dẫn sau:

Đầu tiên, người bán vào trang Seller Center (SC <https://sellercenter.tiki.vn/>) sau đó điều hướng đến

Thông tin người bán (Thông tin nhà bán) > Chi tiết(Xem chi tiết) >
Cài đặt cửa hàng & Thông tin liên hệ (Thiết Lập Cửa Hàng và Thông Tin Liên Lạc) >
Quản lý(Chi tiết) > Ứng dụng tích hợp (Ứng dụng chờ kết nối)



Hình 47 Xác nhận kết nối

3.6.2 Sử dụng Postman để nhận token

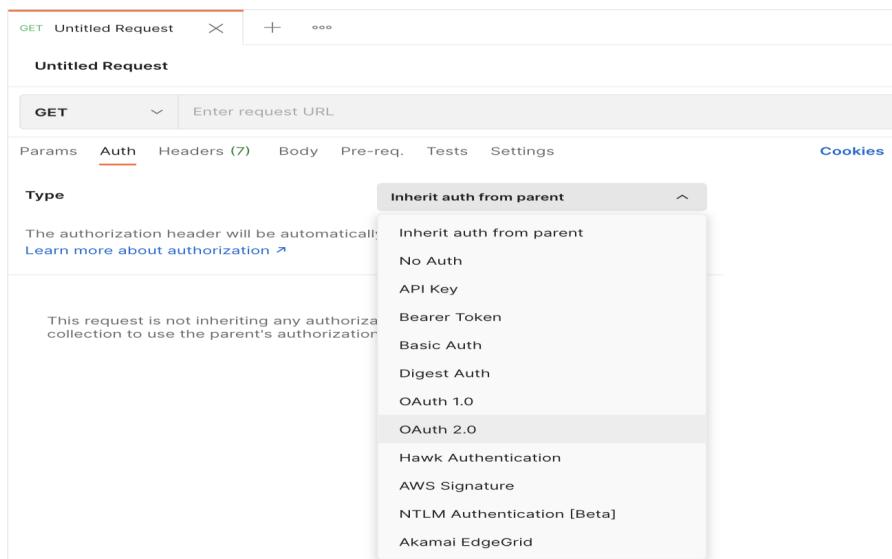
a. Điều kiện tiên quyết

- Có ứng dụng công khai hoặc ứng dụng nội bộ
- Đã có một cửa hàng đang hoạt động trong sản xuất
- Phiên bản Postman $\geq 8.0.0$

b. Ứng dụng công khai được người bán cho phép

Một ứng dụng công khai có thể nhận được mã thông báo truy cập nếu người bán cấp quyền truy cập vào cửa hàng của người bán. Quy trình như sau:

- Tạo yêu cầu Postman
- Mở tab Auth chọn Type chọn OAuth 2.0



Hình 48 Tạo yêu cầu Postman mới

Grant Type phải là **Authorization Code** khi bạn nhận được sự cho phép từ người bán. Và kết quả là:

The screenshot shows the Postman interface with the following configuration for an 'Authorization Code' grant type:

- Method:** GET
- URL:** Untitled Request
- Auth:** (highlighted)
- Headers (7):** Request Headers (with 'Access Token' selected)
- Current Token:** Access Token (Available Tokens: Access Token, Bearer)
- Configure New Token:**
 - Token Name:** Test Client
 - Grant Type:** Authorization Code
 - Callback URL:** {{test-client-redirect-uri}}
 - Auth URL:** {{auth-endpoint}}
 - Access Token URL:** {{token-endpoint}}
 - Client ID:** {{test-client-id}}
 - Client Secret:** {{test-client-secret}}
 - Scope:** order product
 - State:** https://example.com/?a=b
 - Client Authentication:** Send as Basic Auth header

Annotations on the right side of the configuration area:

- Put your desired scope of access here.**
- But only user-approved scopes are granted.**

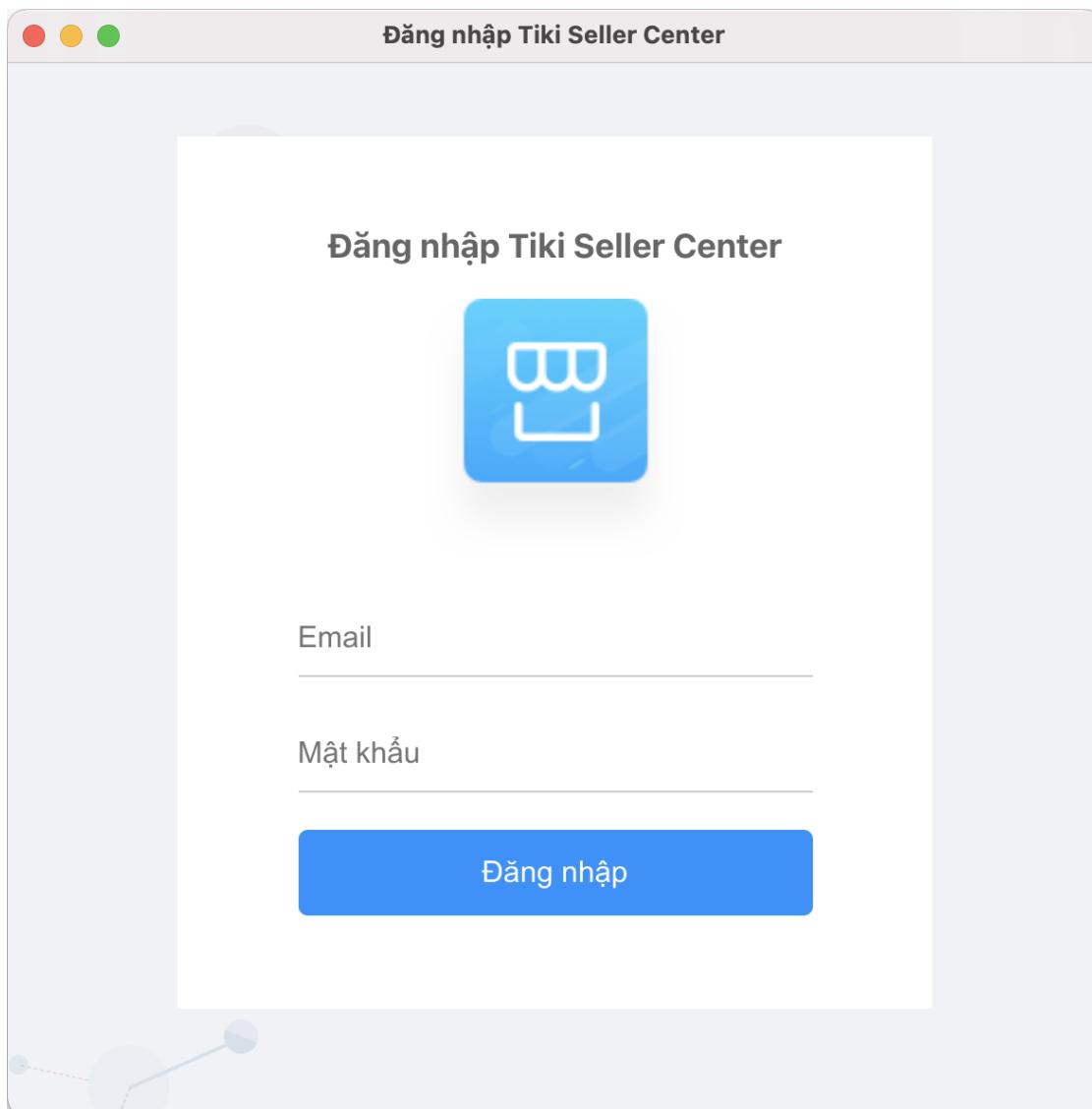
Hình 49 Kết quả

3. Xem lại cấu hình ứng dụng công khai và nhập:

- **Điểm cuối xác thực** và **điểm cuối mã thông báo** từ cấu hình toàn cầu
- **URL chuyển hướng** ứng dụng công khai của bạn

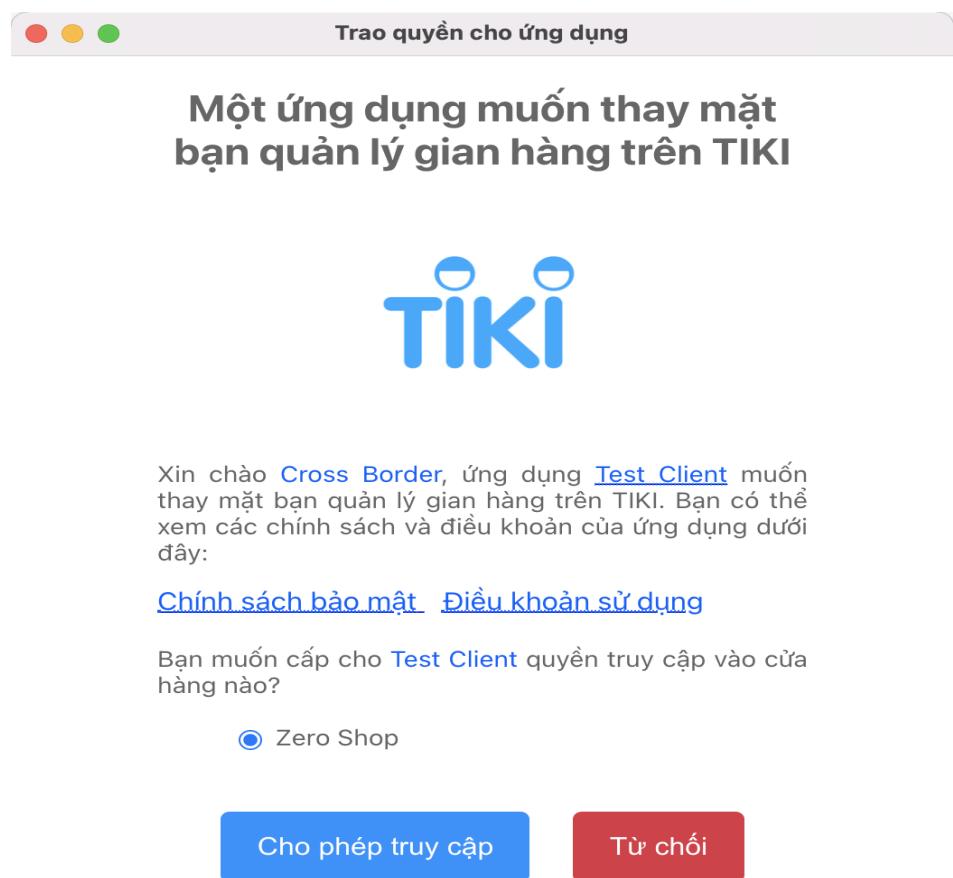
- **ID ứng dụng công khai và bí mật**
- **Phạm vi** muốn yêu cầu từ người bán. Thêm phạm vi offline nếu muốn có một refresh_token.
- **Bất kỳ trạng thái** ngẫu nhiên nào
- **Phương pháp xác thực** ứng dụng công khai của bạn (phương pháp xác thực **điểm cuối mã thông báo**)

4. Nhập vào **Nhận mã thông báo truy cập mới và người bán** sẽ thấy màn hình đăng nhập. Tại đây, bạn phải sử dụng tài khoản **người bán** để đăng nhập vào Trung tâm người bán:



Hình 50 Đăng nhập

5. Người bán nhấp vào **Đăng nhập** (Login) và nhận được màn hình đồng ý. Người bán sẽ chọn 1 trong các cửa hàng của họ có các quyền cần thiết và cấp quyền truy cập ứng dụng của bạn:



Hình 51 Trao quyền cho ứng dụng

6. Người bán nhấp vào **Cho phép truy cập** (Allow) và thì sẽ nhận lại mã thông báo truy cập:

- Sử dụng **access_token** để thực hiện yêu cầu tới **Developer Platform**
- Mã thông báo **access_token** sẽ hết hạn sau **expired_in** vài giây. Có thể sử dụng mã thông báo cho đến khi hết hạn.
- Khi **access_token** hết hạn hoặc bị thu hồi, bất kỳ yêu cầu nào với token đó sẽ dẫn đến lỗi 401 Unauthorized. Nên dựa vào trạng thái http 401 để yêu cầu một token mới hoặc làm mới token bằng **refresh_token**.

Token Details		Use Token
Token Name	Zero Shop → Test Client	
Access Token	wG_4vSa_lKnKte_5zNgZ_yOFtdGbgdv	RBZE8xMp
Token Type	bearer	
expires_in	1295999	
refresh_token	c9anx0NPal-lGiQjiGEPZfda6lUPCC	FCruWh6_7E
scope	order product offline	

Hình 52 Mã thông báo truy cập

c. Ứng dụng nội bộ tự xác thực

Một ứng dụng nội bộ được liên kết với đúng 1 cửa hàng. Ứng dụng chỉ cần xác thực chính nó để có được mã thông báo truy cập.

1. Tạo yêu cầu Postman mới
2. Mở tab **Auth > Type** > Chọn **OAuth 2.0**

The authorization header will be automatically included in all requests. [Learn more about authorization ↗](#)

This request is not inheriting any authorization from its parent collection to use the parent's authorization.

Type

- Inherit auth from parent
- No Auth
- API Key
- Bearer Token
- Basic Auth
- Digest Auth
- OAuth 1.0
- OAuth 2.0** (highlighted)
- Hawk Authentication
- AWS Signature
- NTLM Authentication [Beta]
- Akamai EdgeGrid

Hình 53 Tạo yêu cầu Postman

Grant Type phải là **Client Credentials** khi ứng dụng tự xác thực. Và kết quả là:

The screenshot shows the Postman interface with the following configuration:

- Add auth data to:** Request Headers
- Current Token:** Access Token
- Available Tokens:** Access Token (selected)
- Header Prefix:** Bearer
- Configure New Token:**
 - Token Name:** Dummy Client
 - Grant Type:** Client Credentials
 - Access Token URL:** {{token-endpoint}}
 - Client ID:** {{dummy-client-id}} (with a warning icon)
 - Client Secret:** {{dummy-client-secret}} (with a warning icon)
 - Scope:** (empty) *For in-house apps, leave scope empty.*
 - Client Authentication:** Send as Basic Auth header
- Buttons:** Clear cookies, Get New Access Token

Hình 54 Kết quả

3. Xem lại cấu hình ứng dụng nội bộ và nhập:

- **Điểm cuối mã thông báo** từ cấu hình toàn cầu
- **ID** và **bí mật** ứng dụng nội bộ của bạn
- **Phạm vi** bạn muốn yêu cầu
- **Phương pháp xác thực** ứng dụng nội bộ của bạn (phương pháp xác thực **điểm cuối mã thông báo**)

4. Nhấp vào **Nhận mã thông báo truy cập mới** để xác thực ứng dụng và nhận mã thông báo truy cập (không cần mã thông báo làm mới cho loại cấp Thông tin xác thực của khách hàng).

- Sử dụng **access_token** để thực hiện yêu cầu tới **Developer Platform**
- Mã thông báo **access_token** sẽ hết hạn sau **expired_in** vài giây. Có thể sử dụng mã thông báo cho đến khi hết hạn.

- Khi access_token hết hạn hoặc bị thu hồi, bất kỳ yêu cầu nào với token đó sẽ dẫn đến lỗi 401 Unauthorized. Nên dựa vào trạng thái http 401 để yêu cầu một token mới hoặc làm mới token bằng **refresh_token**.

The screenshot shows a user interface for managing access tokens. On the left, there's a sidebar with 'All Tokens' and a dropdown menu. Below that, it says 'Zero Shop → Test Client' and lists 'Dummy Client'. The main area has a header 'Token Details' with a 'Use Token' button. It displays the following information:

Token Name	Value
Access Token	uiBkBv_sS-5r5_-- 8HbCWY0oGMG
Token Type	bearer
expires_in	1295999
scope	order product

Hình 55 Mã truy cập

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

Sau quá trình tìm hiểu, nghiên cứu và áp dụng giải pháp phù hợp để thu thập dữ liệu từ API của các nền tảng TMĐT. Dựa vào kết quả đạt được tùy thuộc vào mục đích và nhu cầu sử dụng mà có thể chọn ra giải pháp phù hợp.

4.1 Giải pháp thu thập dữ liệu API bằng Python

Ưu điểm:

- **Linh hoạt và tùy biến cao:** python là ngôn ngữ lập trình mạnh mẽ, phổ biến, dễ dàng tùy chỉnh truy vấn API và xử lý dữ liệu theo yêu cầu cụ thể. Điều này giúp các nhà phát triển có thể thay đổi và điều chỉnh các quy trình thu thập dữ liệu tùy vào các API khác nhau và mục đích sử dụng.
- **Thư viện hỗ trợ phong phú:** python có nhiều thư viện mạnh mẽ như requests, pandas, json, asyncio, giúp xử lý API, dữ liệu và tăng tốc quá trình phát triển. Thư viện requests hỗ trợ việc gửi các yêu cầu HTTP đơn giản, trong khi pandas giúp dễ dàng xử lý và phân tích dữ liệu.
- **Tiết kiệm chi phí:** python là mã nguồn mở nên không tốn chi phí. Chỉ cần môi trường lập trình là có thể triển khai hệ thống. Không cần trả phí cho phần mềm bản quyền hoặc giấy phép.
- **Khả năng mở rộng:** có thể mở rộng quy mô một cách linh hoạt để thu thập nhiều nguồn API khác nhau mà không yêu cầu thay đổi quá nhiều trong cấu trúc mã nguồn. Điều này giúp hệ thống có thể phát triển nhanh chóng khi có thêm nhu cầu thu thập dữ liệu từ các nguồn khác.

Nhược điểm:

- **Yêu cầu kỹ năng lập trình:** cần có kiến thức về lập trình Python, API và xử lý dữ liệu. Điều này có thể là một thách thức đối với những người không có kinh nghiệm trong lĩnh vực này.
- **Quản lý môi trường:** phải tự triển khai môi trường chạy mã, đặc biệt là khi làm việc với các hệ thống lớn hoặc dữ liệu yêu cầu tính khả dụng cao. Điều này có thể phức tạp khi xử lý dữ liệu lớn hoặc yêu cầu tính sẵn sàng cao.

- Giới hạn hiệu năng: nếu không tối ưu tốt, tốc độ xử lý API có thể chậm khi lượng dữ liệu lớn hoặc API giới hạn tốc độ gọi (rate limit). Việc xử lý dữ liệu lớn trong bộ nhớ có thể dẫn đến tình trạng chậm hoặc lỗi khi bộ nhớ không đủ.

4.2 Giải pháp thu thập dữ liệu API thông qua phần mềm quản lý đám mây

Ưu điểm:

- Dễ triển khai và sử dụng: các phần mềm quản lý đám mây thường có giao diện đồ họa trực quan, dễ sử dụng và không đòi hỏi quá nhiều kiến thức lập trình. Điều này giúp việc cấu hình và kết nối API trở nên nhanh chóng và thuận tiện, ngay cả đối với những người không có kỹ năng lập trình sâu.

- Hiệu năng cao và ổn định: các nền tảng đám mây như AWS, Google Cloud, Acho hoặc Azure cung cấp tài nguyên phần cứng dồi dào (CPU, RAM, bộ nhớ) và hệ thống lưu trữ ổn định, giúp thu thập và xử lý dữ liệu lớn một cách hiệu quả và không gặp phải tình trạng gián đoạn.

- Khả năng mở rộng: các nền tảng đám mây dễ dàng mở rộng tài nguyên (CPU, RAM,...) khi khôi phục lượng dữ liệu cần xử lý tăng. Việc tăng cường tài nguyên không đòi hỏi sự thay đổi phần cứng hay cơ sở hạ tầng phức tạp.

- Quản lý tự động: phần mềm đám mây thường tích hợp các tính năng tự động hóa như lên lịch thu thập dữ liệu, báo cáo lỗi, sao lưu dữ liệu, và các tác vụ bảo trì. Điều này giúp giảm bớt sự can thiệp của con người trong quá trình vận hành.

- An toàn dữ liệu: phần mềm đám mây cung cấp các công cụ bảo mật mạnh mẽ như mã hóa dữ liệu, sao lưu định kỳ và kiểm soát quyền truy cập, đảm bảo an toàn cho dữ liệu trong suốt quá trình thu thập và xử lý.

Nhược điểm:

- Chi phí cao: sử dụng dịch vụ đám mây thường tốn chi phí, đặc biệt khi xử lý khôi phục lượng dữ liệu lớn hoặc yêu cầu tài nguyên tính toán mạnh mẽ. Chi phí có thể trở nên cao nếu không quản lý hợp lý việc sử dụng.

- Phụ thuộc nhà cung cấp: hệ thống hoạt động phụ thuộc vào dịch vụ của bên thứ ba. Nếu nhà cung cấp gặp sự cố, hệ thống có thể bị gián đoạn hoặc gặp vấn đề về bảo mật, ảnh hưởng đến hoạt động của người sử dụng và của doanh nghiệp.

- **Giới hạn tùy biến:** các phần mềm quản lý đám mây thường được cấu hình cố định, khó tùy chỉnh như lập trình với Python. Khiến cho việc tùy chỉnh hay thay đổi cấu hình phần mềm trở nên khó khăn và hạn chế. Điều này có thể không phù hợp với những yêu cầu đặc thù hoặc phức tạp.
- **Yêu cầu kỹ năng quản lý đám mây:** tuy không đòi hỏi kỹ năng lập trình nhưng vẫn cần phải biết cách sử dụng phần mềm đám mây, để quản lý tài nguyên và chi phí một cách hiệu quả tránh lãng phí.

4.3 Giải pháp thu thập dữ liệu thông qua việc bán hàng

Ưu điểm:

- **Dữ liệu thực tế:** thu thập dữ liệu trực tiếp từ hoạt động bán hàng cung cấp thông tin chính xác về hành vi mua sắm và nhu cầu của khách hàng.
- **Tối ưu hóa chiến lược kinh doanh:** các nhà bán lẻ có thể sử dụng dữ liệu này để phân tích xu hướng tiêu dùng, cải thiện chiến lược marketing và tối ưu hóa tồn kho.
- **Đo lường hiệu quả:** dữ liệu bán hàng cung cấp các chỉ số cụ thể như doanh thu, lượng bán hàng, giúp doanh nghiệp đo lường hiệu quả của các chiến dịch quảng cáo hoặc các thay đổi trong sản phẩm/dịch vụ.
- **Để tiếp cận và cập nhật liên tục:** dữ liệu bán hàng được cập nhật liên tục, giúp doanh nghiệp nắm bắt ngay các thay đổi trong nhu cầu khách hàng.

Nhược điểm:

- **Khó thu thập ở quy mô lớn:** việc thu thập dữ liệu từ hàng loạt giao dịch có thể gặp khó khăn trong việc đồng bộ và quản lý khối lượng lớn thông tin, đặc biệt là từ các cửa hàng hoặc hệ thống bán lẻ khác nhau.
- **Chi phí đầu tư ban đầu:** để thu thập và xử lý dữ liệu bán hàng hiệu quả, doanh nghiệp có thể phải đầu tư vào các hệ thống quản lý điểm bán (POS) hoặc phần mềm chuyên dụng.
- **Bảo mật dữ liệu:** dữ liệu bán hàng có thể chứa thông tin nhạy cảm về khách hàng, vì vậy cần phải có các biện pháp bảo mật mạnh mẽ để bảo vệ quyền riêng tư.

4.4 Giải pháp thu thập dữ liệu thông qua việc hợp tác với các cửa hàng

Ưu điểm:

- **Nguồn dữ liệu phong phú:** hợp tác với nhiều cửa hàng cung cấp dữ liệu từ nhiều nguồn khác nhau, giúp thu thập cái nhìn toàn diện về hành vi người tiêu dùng và xu hướng thị trường.
- **Tiết kiệm chi phí:** việc hợp tác giúp giảm chi phí xây dựng và duy trì hệ thống thu thập dữ liệu riêng biệt. Các cửa hàng đã có sẵn cơ sở hạ tầng và phần mềm cần thiết.
- **Tăng cường mối quan hệ đối tác:** việc hợp tác với các cửa hàng giúp xây dựng mối quan hệ lâu dài, mở rộng mạng lưới thu thập dữ liệu và tạo cơ hội phát triển cho cả hai bên.
- **Dữ liệu thực tế và kịp thời:** dữ liệu thu thập từ các cửa hàng thực tế phản ánh đúng nhu cầu và hành vi của khách hàng trong thời gian gần nhất.

Nhược điểm:

- **Phụ thuộc vào các cửa hàng:** việc thu thập dữ liệu có thể gặp khó khăn nếu các cửa hàng không cung cấp dữ liệu đầy đủ hoặc không duy trì hệ thống thu thập dữ liệu đồng bộ và nếu cửa hàng không có quyền để lấy dữ liệu thì không thể thu thập được dữ liệu của cửa hàng đó.
- **Khó kiểm soát chất lượng dữ liệu:** dữ liệu từ các cửa hàng có thể không đồng nhất hoặc bị sai lệch nếu không có quy trình kiểm tra và xác thực chất lượng.
- **Rủi ro bảo mật và quyền riêng tư:** dữ liệu thu thập từ các cửa hàng có thể chứa thông tin nhạy cảm về khách hàng hoặc sản phẩm, do đó cần phải bảo vệ thông tin này một cách chặt chẽ để tránh vi phạm quyền riêng tư.
- **Khó khăn trong việc đồng bộ hóa dữ liệu:** việc đồng bộ dữ liệu từ nhiều cửa hàng có thể gặp khó khăn, nhất là khi các cửa hàng sử dụng các hệ thống khác nhau để lưu trữ và quản lý dữ liệu.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

Kết quả đạt được:

Các giải pháp thu thập dữ liệu hoạt động thuận lợi, dữ liệu thu thập gồm: đánh giá sản phẩm, đánh giá của khách hàng, đánh giá của hệ thống,..

- **Giải pháp thu thập dữ liệu bằng Python:** phù hợp các lập trình viên, đội ngũ lập trình, những người có khả năng lập trình, yêu cầu tùy chỉnh, tính linh hoạt cao, và ít tốn chi phí.

- **Giải pháp thu thập dữ liệu thông qua phần mềm đám mây:** phù hợp cho các doanh nghiệp, không yêu cầu cao về khả năng lập trình, cần hệ thống thu thập dữ liệu lớn, ổn định, dễ triển khai thực hiện và có khả năng chi trả chi phí khi sử dụng.

- **Giải pháp thu thập dữ liệu qua việc bán hàng:** giải pháp hữu ích cho việc theo dõi hiệu quả sản phẩm và chiến lược marketing, nhưng tính khả thi không cao khi chỉ lấy được dữ liệu của một cửa hàng và yêu cầu sản phẩm phải được bán ra và được khách hàng đánh giá.

- **Giải pháp thu thập dữ liệu thông qua việc hợp tác với các cửa hàng:** giúp thu thập dữ liệu đa dạng hơn so với giải pháp bán hàng, nhưng giải pháp này phụ thuộc vào các cửa hàng.

Những đóng góp mới:

Ứng dụng giải pháp thu thập dữ liệu thông qua API giúp tìm ra giải pháp phù hợp với từng nhu cầu cần thu thập thông tin sản phẩm, người bán và đánh giá từ nhiều nguồn khác nhau.

Đưa ra giải pháp thu thập phù hợp nhằm giảm thiểu công sức và thời gian, nhanh chóng đưa ra được những phương án phát triển phù hợp cho người kinh doanh.

Những đề xuất mới:

Dựa vào những giải pháp trên đưa ra thêm nhiều giải pháp mới hiệu quả hơn, cải tiến hơn, giúp giảm thời gian, công sức, chi phí,... khi muốn thu thập dữ liệu như đánh giá khách hàng, đánh giá người bán, đánh giá hệ thống từ các nền tảng TMĐT.

Hướng phát triển

- **Nghiên cứu các công nghệ tiên tiến:** tích hợp công nghệ để tự động hóa quy trình thu thập, phân tích và dự báo dữ liệu.
- **Mở rộng phạm vi dữ liệu:** thu thập dữ liệu từ nhiều nền tảng khác nhau như mạng xã hội, hệ thống vận chuyển,... để có cái nhìn toàn diện hơn.
- **Phát triển hệ thống:** phát triển hệ thống quản lý dữ liệu để theo dõi và chia sẻ dữ liệu giữa các nền tảng TMĐT, đồng thời hợp tác với các tổ chức để tiêu chuẩn hóa quy trình.
- **Nâng cao hiệu quả triển khai:** tập trung nghiên cứu các giải pháp triển khai nhanh, hiệu quả, giảm chi phí và rủi ro trong việc thu thập và sử dụng dữ liệu.

DANH MỤC TÀI LIỆU THAM KHẢO

[1] Lý thuyết về thương mại điện tử

https://drive.google.com/drive/folders/1uL0vbfmoylQ6wUlPH0b42Gf563I_pO4r

<https://sell.amazon.vn/blog/danh-cho-nguo-i-moi/thuong-mai-dien-tu-la-gi>

[2] Lý thuyết về API

<https://viettelidc.com.vn/tin-tuc/api-la-gi-4-ung-dung-chinh-cua-api-trong-thiet-ke-website-ma-ban-nen-biet>

<https://vina-aspire.com/api-la-gi-tai-sao-api-duoc-su-dung-nhieu-hien-nay/>

<https://fast.com.vn/api-la-gi-nhung-thong-tin-can-biet-ve-api/>

<https://careerviet.vn/vi/talentcommunity/api-la-gi-nhung-dac-diem-noi-bat-cua-api.35A520C1.html#:~:text=API%20l%C3%A0%20vi%E1%BA%BFt%20t%E1%BA%A%A%Ft%20c%E1%BB%A7a%20t%E1%BB%AB%20Application%20Programming%20Interface%20hi%E1%BB%83u,c%C3%B9ng%20c%C3%A1c%20h%C3%A0m%20hay%20d%C3%B9ng.>

<https://viblo.asia/p/tim-hieu-kien-thuc-co-ban-ve-api-maGK7A4Mlj2>

<https://viblo.asia/p/api-la-gi-tat-tan-tat-ve-api-V3m5WXdwKO7>

[3] Lý thuyết về Postman

<https://viblo.asia/p/postman-cong-cu-danh-cho-nguo-i-moi-bat-dau-kiem-thu-api-gAm5ya0w5db>

PHỤ LỤC