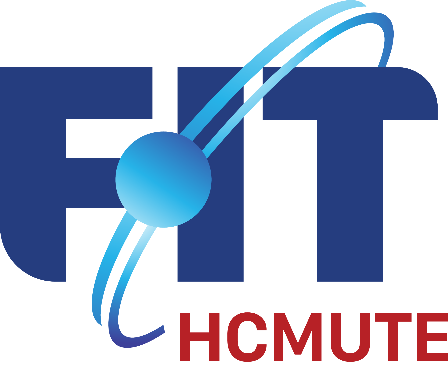
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH KỸ THUẬT DỮ LIỆU**



**LÊ CHÍ HIẾU: 18133012**

**NGUYỄN TUẤN VŨ: 18133064**

Đề tài:

**XÂY DỰNG HỆ THỐNG THEO DÕI HOẠT ĐỘNG WEBSITE (Website Activity Tracking) VỚI APACHE KAFKA**

**TIỂU LUẬN CHUYÊN NGÀNH**

**KHÓA 2018 - 2022**

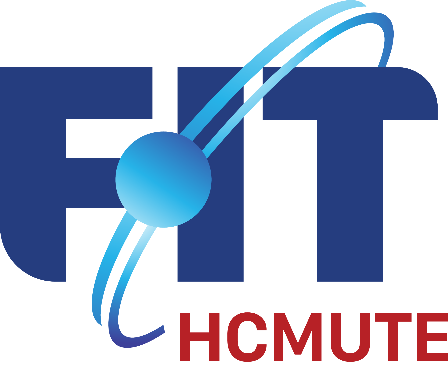
GIÁO VIÊN HƯỚNG DẪN

**ThS. NGUYỄN MINH ĐẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH KỸ THUẬT DỮ LIỆU**



**KHÓA 2018 - 2022**

**LÊ CHÍ HIẾU: 18133012**

**NGUYỄN TUẤN VŨ: 18133064**

GIÁO VIÊN HƯỚNG DẪN

**ThS. NGUYỄN MINH ĐẠO**

Đề tài:

**XÂY DỰNG HỆ THỐNG THEO DÕI HOẠT ĐỘNG WEBSITE (Website Activity Tracking) VỚI APACHE KAFKA**

**TIỂU LUẬN CHUYÊN NGÀNH**

|  |  |  |
| --- | --- | --- |
| **ĐH SƯ PHẠM KỸ THUẬT TP.HCM**  **KHOA CNTT**  **\*\*\*\*\*\*\*\*** | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh Phúc**  **\*\*\*\*\*\*\*\*** | |
| **PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN** | | |
| Họ và tên Sinh viên 1: Lê Chí Hiếu  Họ và tên Sinh viên 2: Nguyễn Tuấn Vũ | | MSSV: 18133012  MSSV: 18133064 |
| Ngành: Kỹ thuật dữ liệu  Tên đề tài: XÂY DỰNG HỆ THỐNG THEO DÕI HOẠT ĐỘNG WEBSITE (Website Activity Tracking) VỚI APACHE KAFKA  Họ và tên giáo viên phản biện: Nguyễn Minh Đạo | | |
| **NHẬN XÉT**  1. Về nội dung đề tài khối lượng thực hiện: ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 2. Ưu điểm: ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 3. Khuyết điểm:  ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 4. Đề nghị cho bảo vệ hay không? 5. Đánh giá loại:  6. Điểm: | | |
| Tp*. Hồ Chí Minh, ngày    tháng     năm   2021*  Giáo viên hướng dẫn  *(Ký & ghi rõ họ tên)* | | |
| **ĐH SƯ PHẠM KỸ THUẬT TP.HCM**  **KHOA CNTT**  **\*\*\*\*\*\*\*\*** | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh Phúc**  **\*\*\*\*\*\*\*\*** | |
| **PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN** | | |
| Họ và tên Sinh viên 1: Lê Chí Hiếu  Họ và tên Sinh viên 2: Nguyễn Tuấn Vũ | | MSSV: 18133012  MSSV: 18133064 |
| Ngành: Kỹ thuật dữ liệu  Tên đề tài: XÂY DỰNG HỆ THỐNG THEO DÕI HOẠT ĐỘNG WEBSITE (Website Activity Tracking) VỚI APACHE KAFKA  Họ và tên giáo viên phản biện: Quách Đình Hoàng | | |
| **NHẬN XÉT**  1. Về nội dung đề tài khối lượng thực hiện: ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 2. Ưu điểm: ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 3. Khuyết điểm:  ………………………………………………………………………………………………………  ………………………………………………………………………………………………………  ……………………………………………………………………………………………………… | | |
| 4. Đề nghị cho bảo vệ hay không? 5. Đánh giá loại:  6. Điểm: | | |
| Tp*. Hồ Chí Minh, ngày    tháng     năm   2021*  Giáo viên hướng dẫn  *(Ký & ghi rõ họ tên)* | | |

**LỜI CAM ĐOAN**

Tiểu luận này là công trình nghiên cứu của chúng tôi, được thực hiện dưới sự hướng dẫn của thầy Nguyễn Minh Đạo. Các số liệu, những kết luận nghiên cứu và sản phẩm được tạo ra bởi chúng tôi được trình bày trong khóa luận này là trung thực.

Chúng tôi xin hoàn toàn chịu trách nhiệm về lời cam đoan này.

|  |  |
| --- | --- |
| Sinh viên thực hiện  (Ký và ghi rõ họ tên) | Sinh viên thực hiện  (Ký và ghi rõ họ tên) |
| **Lê Chí Hiếu** | **Nguyễn Tuấn Vũ** |

**LỜI CẢM ƠN**

Trước hết chúng tôi xin cảm ơn đến thầy Nguyễn Minh Đạo là giảng viên hướng dẫn của chúng tôi. Mặc dù tình hình dịch bệnh phức tạp thầy cũng hướng dẫn và cho chúng tôi những tài liệu để hoàn thành tiểu luận chuyên ngành đúng theo thời gian quy định. Qua đây chúng tôi cũng gửi lời cảm ơn chân thành đến Thầy Cô khoa Công nghệ Thông tin - Đại học Sư phạm Kỹ thuật TP.HCM đã đồng hành và hỗ trợ chúng tôi trong suốt quá trình học tập và thực hiện tiểu luận. Chúng tôi xin cảm ơn trường Đại học Sư phạm Kỹ thuật đã tạo nhiều điều kiện thuận lợi cho hoạt động phục vụ học tập của sinh viên chúng tôi, đặc biệt là thư viện số với nguồn tri thức vô tận. Chúng tôi cũng gửi lời cảm ơn chân thành đến các bạn khóa 2018 ngành Kỹ thuật Dữ liệu, cảm ơn những góp ý và chia sẻ quý giá từ tất cả các bạn. Cảm ơn sự động viên từ các bạn để nhóm chúng tôi có thể giữ vững tinh thần và thực hiện khóa luận đúng tiến độ.

Tp. Hồ Chí Minh, ngày … tháng 01 năm 2022

Nhóm sinh viên thực hiện

LÊ CHÍ HIẾU 18133012

NGUYỄN TUẤN VŨ 18133064

**MỤC LỤC**

[PHẦN 1: PHẦN MỞ ĐẦU 15](#_Toc93160450)

[1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI 15](#_Toc93160451)

[2. MỤC TIÊU VÀ NHIỆM VỤ NGHIÊN CỨU 15](#_Toc93160452)

[3. CÁCH TIẾP CẬN VÀ PHƯƠNG PHÁP NGUYÊN CỨU 16](#_Toc93160453)

[4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC 16](#_Toc93160454)

[5. BỐ CỤC TIỂU LUẬN 16](#_Toc93160455)

[PHẦN 2: PHẦN NỘI DUNG 17](#_Toc93160456)

[CHƯƠNG 1: TỔNG QUAN VỀ WEB TRACKING 17](#_Toc93160457)

[1.1 Web tracking là gì? 17](#_Toc93160458)

[1.1.1 Các bên sử dụng, vận hành Web tracking: First-Party, Third-Party, Fourth-Party Tracking. 17](#_Toc93160459)

[1.1.2 Lợi ích của Web tracking 18](#_Toc93160460)

[1.2 Các kỹ thuật mà các trang web sử dụng để theo dõi hoạt động của người dùng 19](#_Toc93160461)

[**1.2.1** **IP address tracking** 19](#_Toc93160462)

[**1.2.2** **HTTP cookies** 19](#_Toc93160463)

[**1.2.3** **Pixel tracking hay web beacons** 20](#_Toc93160464)

[**1.2.4** **Browser fingerprinting** 21](#_Toc93160465)

[**1.2.5** **Canvas fingerprinting** 21](#_Toc93160466)

[**1.2.6** **Device Tracking** 22](#_Toc93160467)

[**1.3** **Một số use case sử dụng web tracking** 22](#_Toc93160468)

[**1.3.1** **Ecommerce** 22](#_Toc93160469)

[**1.3.2** **Website analytics** 22](#_Toc93160470)

[**1.3.3** **Advertising** 23](#_Toc93160471)

[**1.3.4** **Usability testing** 23](#_Toc93160472)

[**1.4** **Các vấn đề về pháp lý, tính hợp pháp** 24](#_Toc93160473)

[1.4.1 Quy định chung về bảo vệ dữ liệu - General Data Protection Regulation (GDPR) 24](#_Toc93160474)

[1.4.2 Đạo luật Quyền riêng tư của Người tiêu dùng California - California Consumer Privacy Act (CCPA) 24](#_Toc93160475)

[1.4.3 Các văn bản pháp luật quy định về bảo vệ dữ liệu, thông tin cá nhân và quyền riêng tư tại Việt Nam 25](#_Toc93160476)

[CHƯƠNG 2 TỔNG QUAN VỀ HỆ THỐNG STREAMING 27](#_Toc93160477)

[2.1 Hệ thống streaming 27](#_Toc93160478)

[2.1.1 Hệ thống real-time 27](#_Toc93160479)

[2.1.2 Hệ thống streaming là gì? 27](#_Toc93160480)

[2.1.3 Lý do đề cập đến hệ thống streaming? 28](#_Toc93160481)

[2.1.4 Kiến trúc 1 hệ thống streaming: 28](#_Toc93160482)

[2.2 Message queue Tier 30](#_Toc93160483)

[2.2.1 Message queue 30](#_Toc93160484)

[2.2.1.1 Message queue là gì? 30](#_Toc93160485)

[2.2.1.2 Thành phần của một message queue 30](#_Toc93160486)

[2.2.1.3 Lợi ích của message queue trong web tracking 31](#_Toc93160487)

[2.2.2 Apache Kafka 32](#_Toc93160488)

[2.2.2.1 Event streaming là gì? 32](#_Toc93160489)

[2.2.2.2 Apache Kafka là gì? 32](#_Toc93160490)

[2.2.2.3 Cách thức hoạt động. 32](#_Toc93160491)

[2.2.2.4 Ưu điểm của Kafka 33](#_Toc93160492)

[2.3 Analysis tier 34](#_Toc93160493)

[2.3.1 Spark 34](#_Toc93160494)

[2.3.1.1 Hoàn cảnh ra đời 34](#_Toc93160495)

[2.3.1.2 Kiến trúc Spark 35](#_Toc93160496)

[2.3.1.3 Cơ chế hoạt động 36](#_Toc93160497)

[2.3.2 Spark Streaming 37](#_Toc93160498)

[2.3.2.1 Spark Streaming là gì? 37](#_Toc93160499)

[2.3.2.2 Cách hoạt động: 37](#_Toc93160500)

[2.3.2.3 Bốn ưu điểm chính của Spark Streaming 38](#_Toc93160501)

[CHƯƠNG 3. TỔNG QUAN VỀ MERN STACK 39](#_Toc93160502)

[3.1 React.JS 39](#_Toc93160503)

[3.2 Express.JS và Node.JS 40](#_Toc93160504)

[3.3 MongoDB 40](#_Toc93160505)

[3.4 Hoạt động của MERN Stack 40](#_Toc93160506)

[3.5 Ưu điểm của MERN Stack 41](#_Toc93160507)

[3.5.1 Thiết kế use case 41](#_Toc93160508)

[3.5.2 Có thể code Isomorphic 41](#_Toc93160509)

[3.5.3 Tính linh hoạt cao 41](#_Toc93160510)

[3.5.4 Tối ưu chi phí 41](#_Toc93160511)

[3.5.5 Có tốc độ cao và tái sử dụng được 41](#_Toc93160512)

[3.5.6 Mã nguồn mở và có khả năng tương thích với đám mây 41](#_Toc93160513)

[CHƯƠNG 4. Thiết kế, cài đặt và kiểm thử 43](#_Toc93160514)

[4.1 Sơ đồ hoạt động 43](#_Toc93160515)

[4.2 Môi trường cài đặt và các công nghệ liên quan 44](#_Toc93160516)

[**4.2.1** **Môi trường cài đặt web** 44](#_Toc93160517)

[4.2.2 MongoDB 44](#_Toc93160518)

[**4.2.3** **Cài đặt môi trường cho Apache Hadoop, Apache Spark, Apache Kafka và** 49](#_Toc93160519)

[4.3 Thiết lập mô hình đề xuất các sản phẩm tương tự với Root mean square error. 51](#_Toc93160520)

[4.4 Quy trình hoạt động của hệ thống Web tracking: 52](#_Toc93160521)

[PHẦN 3: PHẦN KẾT LUẬN 57](#_Toc93160522)

[Kết quả đạt được 57](#_Toc93160523)

[Hạn chế 57](#_Toc93160524)

[Hướng phát triển 57](#_Toc93160525)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Hệ thống kiến trúc Streaming 29](#_Toc92226735)

[Hình 2: Thành phần của một Message queue 30](#_Toc92226736)

[Hình 3: Cách thức hoạt động của Broker 33](#_Toc92226737)

[Hình 4: Ưu điểm của Apache Kafka [14] 33](#_Toc92226738)

[Hình 5: Hệ thống Apache Spark 35](#_Toc92226739)

[Hình 6: Kiến trúc Spark 35](#_Toc92226740)

[Hình 7: Spark Streaming [15] 37](#_Toc92226741)

[Hình 8. Spark Streaming Pipeline [15]. 38](#_Toc92226742)

[Hình 9: Kiến trúc của MERN STACK 39](#_Toc92226743)

[Hình 10: Quy trình hoạt động của MERN STACK 40](#_Toc92226744)

[Hình 11: Sơ đồ hoạt động 43](#_Toc92226745)

[Hình 12: Phiên bản của node 44](#_Toc92226746)

[Hình 13: Phiên bản npm 44](#_Toc92226747)

[Hình 14: Giao diện MongoDB 44](#_Toc92226748)

[Hình 15: Database và Collections 45](#_Toc92226749)

[Hình 16: Môi trường Hadoop 49](#_Toc92226750)

[Hình 17: Môi trường Spark 49](#_Toc92226751)

[Hình 18: Môi trường Kafka 50](#_Toc92226752)

[Hình 19: Môi trường Zookeeper 50](#_Toc92226753)

[Hình 20: Thêm thư viện spark-default.conf 51](#_Toc92226754)

[Hình 21: Dữ liệu sau khi tiền xử lý 51](#_Toc92226755)

[Hình 22: Dữ liệu sau khi StandardScaler 52](#_Toc92226756)

[Hình 23: Hàm tính toán mức độ tương đồng 52](#_Toc92226757)

[Hình 24: Giao diện web 53](#_Toc92226758)

[Hình 25: Chi tiết sản phẩm 54](#_Toc92226759)

[Hình 26: \_id gửi xuống nodejs 54](#_Toc92226760)

[Hình 27: Danh sách sản phẩm gợi ý 55](#_Toc92226761)

[Hình 28: Danh sách mã sản phẩm 55](#_Toc92226762)

[Hình 29: Gợi ý 5 sản phẩm 56](#_Toc92226763)

**DANH MỤC BẢNG BIỂU**

[Bảng 1: Phân loại hệ thống Real-time 27](#_Toc92226460)

[Bảng 2: Thuộc tính bảng Users 45](#_Toc92226461)

[Bảng 3: Thuộc tính bảng Products 46](#_Toc92226462)

[Bảng 4: Thuộc tính bảng Trackings 47](#_Toc92226463)

[Bảng 5: Thuộc tính bảng Similarities 48](#_Toc92226464)

[Bảng 6: Thuộc tính bảng Payments 48](#_Toc92226465)

[Bảng 7: Thuộc tính bảng Categories 49](#_Toc92226466)

**DANH MỤC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **WEB** | Trang web |
| **DOM** | Document Object Model |
| **HTML** | Hyper Text Markup Language |
| **JSON** | JavaScript Object Notation |
| **JS** | JavaScript |
| **XML** | Extensible Markup Language |

# PHẦN 1: PHẦN MỞ ĐẦU

## 1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Những năm gần đây, “thương mại điện tử” (TMĐT) đã không còn là khái niệm xa lạ trong xã hội hay một lĩnh vực mới mẻ tại nước ta. Có thể coi năm 2020, đại dịch COVID-19 đã mang đến nhiều biến động đối với nền kinh tế và sự tăng trưởng bứt phá của TMĐT đã góp phần đưa Việt Nam trở thành một trong những thị trường tiềm năng nhất khu vực ASEAN [1]. Với lượng người dùng đang trên đà tăng trưởng chóng mặt vì tình hình dịch bệnh thì việc nắm bắt hành vi, thói quen, sở thích của người dùng nhằm tối ưu trải nghiệm, tối ưu quảng cáo hướng đối tượng, tối ưu các đề xuất sản phẩm phù hợp với nhu cầu của người dùng nhằm nâng cao doanh thu, lợi nhuận của các sàn thương mại điện tử mà vẫn đảm bảo tuân thủ pháp luật là ngày càng quan trọng và cấp thiết. Và để đáp ứng được việc xử lý dữ liệu của hàng triệu sự kiện trên mỗi giây mà người dùng tạo ra trên sàn thương mại điện tử, nhóm đề xuất sử dụng công cụ Apache Kafka và đề xuất đề xuất “Tìm hiểu và xây dựng hệ thống theo dõi hoạt động website (Website Activity Tracking) với Apache Kafka” làm đề tài của mình.

## 2. MỤC TIÊU VÀ NHIỆM VỤ NGHIÊN CỨU

Mục tiêu: Thiết lập hệ thống web tracking theo dõi hoạt động click sản phẩm của người dùng và gợi ý những sản phẩm tương tự sản phẩm người dùng vừa click theo thời gian thực.

Nhiệm vụ:

* Tìm hiểu khái niệm web tracking, phân loại kỹ thuật, các trường hợp sử dụng, lợi ích.
* Tìm hiểu về các vấn đề pháp lý về thông tin cá nhân quyền riêng tư, các điều kiện thu thập lưu trữ phân tích dữ liệu cá nhân một cách hợp pháp.
* Tìm hiểu công nghệ Apache Kafka, Apache Spark Streaming để hỗ trợ thu thập xử lý dữ liệu thời gian thực.
* Tìm hiểu công nghệ MERN Stack gồm: NodeJS, ExpressJS, ReactJS, MongoDB.

## 3. CÁCH TIẾP CẬN VÀ PHƯƠNG PHÁP NGUYÊN CỨU

Cách tiếp cận:

* + - * Sử dụng lí thuyết về Web tracking, hệ thống Streaming.
* Sử dụng các công nghệ Apache Kafka, Apache Spark Streaming, MongoDB, ExpressJS, ReactJS, NodeJS.

## 4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC

* Về mặt lí thuyết: Thực nghiệm được một kĩ thuật Web tracking là advertising.
* Về mặt sản phẩm: Thiết lập hệ thống Web tracking theo dõi hoạt động Click sản phẩm của người dùng và gợi ý những sản phẩm tương tự sản phẩm người dùng vừa Click theo thời gian thực.

## 5. BỐ CỤC TIỂU LUẬN

Chúng tôi định chia bố cục tiểu luận thành các phần như sau:

* Phần 1: Mở đầu
* Phần 2: Nội dung. Phần này gồm 4 chương
  + Chương 1: Tổng quan về Web tracking
  + Chương 2: Tổng quan về hệ thống Streaming data
  + Chương 3: Tổng quan về MERN Stack
  + Chương 4: Thiết kế, cài đặt, kiểm thử hệ thống Web tracking
* Phần 3: Kết luận
* Tài liệu tham khảo

# PHẦN 2: PHẦN NỘI DUNG

# CHƯƠNG 1: TỔNG QUAN VỀ WEB TRACKING

## Web tracking là gì?

Web tracking là một cách thức, kỹ thuật được sử dụng để thu thập, lưu trữ và phân tích hoạt động của người dùng trên một hoặc một số trang web. Khi việc thu thập, lưu trữ, phân tích hoạt động của người dùng được thực hiện đúng, việc theo dõi trang web và người dùng có thể mang lại lợi ích cho cả người dùng cá nhân và doanh nghiệp.

### 1.1.1 Các bên sử dụng, vận hành Web tracking: First-Party, Third-Party, Fourth-Party Tracking.

* **First-Party Tracking:** Theo dõi bên thứ nhất được thực hiện bởi trang web người dùng đã chọn để truy cập. Ví dụ: khi một cửa hàng thương mại điện tử theo dõi các trang bạn đã truy cập để cung cấp cho bạn mục “Recently Viewed” hiển thị một số mặt hàng bạn đã xem trước đây. [2].

Theo dõi của bên thứ nhất được thực hiện bởi trang web mà người dùng đang truy cập. Theo dõi khách truy cập trang web thuộc loại này giám sát hành vi của người dùng để ghi nhớ các tùy chọn của họ, chẳng hạn như loại nội dung người dùng thường thích, cũng như cài đặt ngôn ngữ, vị trí và thông tin khác mà người dùng chia sẻ. Thông thường, người dùng không có gì phải lo lắng về việc theo dõi tùy chọn của bên thứ nhất, vì người dùng có thể muốn các trang web mà họ truy cập thường xuyên hiển thị các tùy chọn, thông tin phù hợp với sở thích, nhu cầu của họ, … (Tình hình trở nên tồi tệ hơn nếu các trang web của bên thứ nhất bán quyền truy cập vào dữ liệu của người dùng cho các nhà tiếp thị, quảng cáo).

* **Third-Party Tracking:** Theo dõi của bên thứ ba xảy ra khi các trang web khác ngoài trang web người dùng đang sử dụng cũng theo dõi hoạt động của người dùng. Ví dụ: người dùng có thể truy cập một trang web tin tức để đọc nội dung của họ mà không nhận ra rằng trang web đó cũng tải một loạt các cookie của bên thứ ba để theo dõi hành vi của người dùng cả trên trang web của họ cũng như các trang web khác mà người dùng có thể truy cập trong tương lai [2].

Đó là lý do tại sao định nghĩa về trang web của bên thứ ba bao gồm tất cả các trang web có thể xem bạn đang làm gì trên các trang web khác. Việc theo dõi hành vi của bên thứ ba thường này được sử dụng để giúp các nhà quảng cáo điều chỉnh quảng cáo của họ cho phù hợp với sở thích, nhu cầu của người dùng**.**

* **Fourth-Party Tracking:** Có thể được hiểu là những đoạn code, script tracking của bên phụ trợ, hỗ trợ, cung cấp dịch vụ cho bên thứ 3. Ví dụ: Nhà phát triển web A sử dụng embedded YouTube clip để giới thiệu sản phẩm trên trang web của mình. Tuy nhiên, Google cũng có cài đặt pixel tracking Double-Click trên trang web của A. YouTube thuộc về Google và Google sử dụng nền tảng quảng cáo của mình, Double-Click, để theo dõi ai đang xem video YouTube của A. Như vậy, bây giờ A có một pixel tracking Double Click thuộc bên thứ tư trên trang web của mình [3].

### 1.1.2 Lợi ích của Web tracking

* Web tracking giúp ích gì cho người dùng?

Web tracking có thể giúp ghi nhớ thông tin đăng nhập và tùy chọn, do đó người dùng không cần phải đăng nhập lại mỗi khi bạn truy cập một trang web, hoặc người dùng đã thêm một loạt các sản phẩm vào giỏ hàng trên một trang thương mại điện tử của mình nhưng không thanh toán, các sản phẩm vẫn có thể được lưu trữ cho người dùng vài ngày sau đó.

Tương tự, Web tracking có thể giúp các trang web và đề xuất sản phẩm mang lại trải nghiệm được cá nhân hóa hơn, chẳng hạn như điều chỉnh nội dung dựa trên vị trí hoặc múi giờ của khách truy cập tạo ra trải nghiệm trang web và sản phẩm thú vị và hữu ích hơn.

* Web tracking giúp ích như thế nào cho các doanh nghiệp?

Một trong những lý do chính mà các trang web theo dõi người dùng là để có được cái nhìn sâu sắc về cách khách truy cập sử dụng trang web của họ, cung cấp trải nghiệm người dùng được cá nhân hóa và kiếm tiền từ người dùng bằng cách hiển thị cho họ các quảng cáo phù hợp.

Cải thiện trải nghiệm người dùng không chỉ mang lại lợi ích cho khách hàng: người dùng hài lòng còn có lợi cho việc kinh doanh, dẫn đến cải thiện các chỉ số như CTR (Click-Through Rate), giảm tỷ lệ thoát cũng như tăng lượt đăng ký và bán hàng.

Web tracking cũng giúp các nhà phát triển đo lường hiệu suất của các chiến dịch Marketing, bao gồm cả hiệu suất của Search Engine Optimization, cũng như có được thông tin chi tiết có giá trị để giúp đưa ra quyết định [4].

## Các kỹ thuật mà các trang web sử dụng để theo dõi hoạt động của người dùng

* + 1. **IP address tracking**

Mỗi máy tính trên Internet đều có địa chỉ IP duy nhất (Internet Protocol), được sử dụng để định tuyến dữ liệu tới nó [5].

Địa chỉ IP của người dùng cho biết vị trí vật lý của họ, các trang web sử dụng IP tracking để theo dõi nơi khách truy cập của họ đến. Họ cũng có thể sử dụng địa chỉ IP để xác định các kiểu hành vi và xuất các sản phẩm phù hợp với người dân vùng miền đó.

IP tracking chỉ hoạt động nếu địa chỉ IP của người dùng có thể lấy được. Nếu người dùng ẩn địa chỉ IP của mình bằng proxy hoặc VPN, thì phương pháp theo dõi này sẽ kém hiệu quả [2]**.**

* + 1. **HTTP cookies**

Cookie là những phần dữ liệu nhỏ mà các trang web lưu trữ trên thiết bị của người dùng. Các trang web thường sử dụng cookie để ghi nhớ sở thích của người dùng và mang lại trải nghiệm được cá nhân hóa, cũng như để lấy thông tin cho quảng cáo.

Sau khi trang web thả cookie trên máy tính của người dùng, nhà cung cấp cookie có thể tiếp tục truy cập vào nó. Đây là cách các trang web có thể sử dụng cookie để theo dõi người dùng từ trang này sang trang khác trong cùng 1 trang web hoặc từ trang web này này sang trang web khác.

Cookie của bên thứ nhất được tạo bởi trang web bạn đang truy cập và chúng chủ yếu giúp trang web đó ghi nhớ những gì bạn thích và những gì bạn đang làm. Cookie của bên thứ nhất lưu có thể lưu nhiều loại thông tin như múi giờ người dùng đang sống và thông tin đăng nhập của họ. Trong hầu hết các trường hợp, cookie của bên thứ nhất là cần thiết cho chức năng của trang web hoặc ít nhất là hữu ích trong việc cải thiện trải nghiệm người dùng trên trang web [2].

Cookie của bên thứ ba được tạo bởi các bên khác với trang web người dùng đang truy cập. Đây là những cookie theo dõi theo dõi người dùng khi họ duyệt từ trang web này sang trang web tiếp theo. Website analytics và advertising là hai trong số những cách sử dụng chính để theo dõi cookie - trên thực tế, một nghiên cứu gần đây cho thấy rằng 99% tất cả cookie là theo dõi và cookie quảng cáo [7].

Thời gian theo dõi người dùng của cookie tùy thuộc vào loại cookie.

Session cookies chỉ được lưu trữ trên máy tính của người dùng trong suốt session (phiên làm việc) của họ; chúng biến mất khi người dùng đóng trình duyệt. Persistent cookies là những cookie có tuổi thọ do trang web đặt ra. Những cookie này có thể được đặt để ở trên máy tính trong nhiều năm hoặc tới khi người dùng xóa chúng[6]**.**

* + 1. **Pixel tracking hay web beacons**

Pixel tracking hoặc tên gọi khác là web beacons là một tag nhỏ được đặt trên trang web hoặc trong email để theo dõi cách người dùng tương tác với nội dung. Những web beacons thường là hình ảnh trong suốt có kích thước 1 pixel x 1pixel được đặt trong mã của trang web.

Khi trình duyệt truy cập vào một trang web có web beacons, trình duyệt sẽ gửi request tải xuống hình ảnh đã nêu trên. Request sẽ chứa các chi tiết có thể theo dõi người dùng bao gồm địa chỉ IP, thời gian hoặc thông tin về trình duyệt.

Điều này cho phép quản trị viên web theo dõi người dùng khi chuyển hướng sang một trang khác. Ngoài ra, khi được sử dụng trong email marketing, chúng có thể cung cấp cho công ty đã gửi thông tin email biết được rằng người dùng đã đọc email hay chưa và đọc bao nhiêu lần.

Trên các trang web, web beacons hoạt động giống như cookie ở chỗ chúng được sử dụng để theo dõi người dùng cho các mục đích phân tích trang web và quảng cáo. Các beacons này giám sát cách người sử dụng một trang hoặc điều hướng qua một loạt trang và các công ty sử dụng thông tin này để điều chỉnh các dịch vụ của họ cho phù hợp với hành vi và sở thích của người dùng [6]**.**

* + 1. **Browser fingerprinting**

Browser Fingerprinting là một hình thức theo dõi trang web sử dụng các thuộc tính của thiết bị hoặc trình duyệt của người dùng để xây dựng hồ sơ (Profiling) của người dùng. Thông tin mà Fingerprinting thu thập có thể bao gồm thiết bị truy cập, Laptop, PC, hệ điều hành, độ phân giải màn hình, trình duyệt và phiên bản trình duyệt, ngôn ngữ, plugins, múi giờ và các loại thông tin khác.

Nếu để riêng, mỗi phần thông tin dường như không có giá trị. Tuy nhiên, khi tất cả được kết hợp với nhau, nó cung cấp một cách cực kỳ chính xác để xác định người dùng. Vì khả năng rất nhỏ là hai người dùng bất kỳ có cùng dữ liệu trình duyệt giống hệt nhau, bộ thông số này sẽ trở thành “browser fingerprint” (tạm dịch: dấu vân tay trình duyệt) của người dùng[6]**.**

* + 1. **Canvas fingerprinting**

HTML5 - phiên bản mới nhất của ngôn ngữ ngôn ngữ đánh dấu ssiêu văn bản HTML mà các nhà phát triển sử dụng để xây dựng trang web - bao gồm phần tử “canvas”. Ban đầu được sử dụng để vẽ đồ họa và hoạt ảnh trên một trang web, giờ đây nó đã được chọn trở thành một công cụ lấy dấu vân tay (fingerprinting) mạnh mẽ.

Khi bạn truy cập một trang web sử dụng canvas fingerprinting, trang web đó sẽ hướng dẫn trình duyệt vẽ một hình ảnh ẩn. Do sự khác nhau về phần cứng thiết bị, card đồ họa và cài đặt của mỗi người, trình duyệt mỗi người dùng sẽ hiển thị hình ảnh hơi khác so với trình duyệt của người khác, cung cấp thông tin về vân tay kỹ thuật số duy nhất của bạn. Khi được ghép nối với thông tin theo dõi khác, canvas fingerprinting có độ chính xác cao.

Do có nhiều mục đích sử dụng hợp pháp phần tử canvas HTML5 và không giống như cookie mà người dùng có thể dễ dàng chặn hoặc xóa, canvas fingerprinting là một trong những phương pháp theo dõi mạnh mẽ và khó bị loại bỏ nhất[2]**.**

* + 1. **Device Tracking**

Dữ liệu thiết bị — bao gồm hệ điều hành, trình duyệt và loại thiết bị (tức là máy tính để bàn, laptop hoặc thiết bị di động) - được thu thập bởi nhiều công cụ phân tích và có thể được sử dụng để khắc phục lỗi trang web và các vấn đề thiết kế trên các thiết bị, trình duyệt hoặc độ phân giải màn hình cụ thể.

* 1. **Một số use case sử dụng web tracking** 
     1. **Ecommerce**

Các trang web thương mại điện tử sử dụng web tracking để cố gắng bán cho người dùng nhiều sản phẩm hơn. Mọi thứ người dùng làm trên trang web thương mại điện tử đều được ghi lại, đo lường và phân tích liên tục để tối ưu hóa trang web và thúc đẩy doanh số bán hàng.

Các trang web thương mại điện tử tìm hiểu về thói quen, sở thích và phong cách mua sắm tổng thể. Sau đó, họ sử dụng dữ liệu đó để chuyển đổi người dùng thành khách hàng.

Nhiều trang web thương mại điện tử sử dụng phần mềm của bên thứ ba được gọi là nền tảng dữ liệu khách hàng (customer data platforms) để xử lý dữ liệu khách hàng của họ và sử dụng thông tin thu thập được để tạo các chiến dịch email, quảng cáo hướng mục tiêu, thúc đẩy doanh số bán hàng. Có cả ngành công nghiệp về nền tảng dữ liệu khách hàng kiếm và tiền từ việc giúp các trang thương mại điện tử chuyển “thông tin thu thập từ người dùng” của khách hàng thành doanh số bán hàng [2]**.**

* + 1. **Website analytics**

Theo dõi người dùng cho các mục đích phân tích là vô cùng phổ biến. Công cụ phân tích phổ biến nhất là Google Analytics, công cụ theo dõi trang web được công ty xây dựng ước tính đang sử dụng trên 28 triệu trang web, bao gồm khoảng 74.8% trong số 10.000 trang hàng đầu và khoảng 69.77% trong số 100.000 trang hàng đầu [8].

Các trang web thường sử dụng phần mềm phân tích để thu thập thông tin về khách hàng của họ. Điều này có thể bao gồm nhân khẩu học của khách truy cập trang web và cách những khách hàng này sử dụng trang web. Ví dụ: cách họ đến trang web và họ truy cập bao nhiêu trang.

Thông tin này có thể giúp chủ sở hữu trang web đưa ra quyết định kinh doanh và tối ưu hóa trang web dựa trên cách khách truy cập sử dụng nó [6]**.**

* + 1. **Advertising**

Các trang web thu thập dữ liệu về người dùng để cung cấp cho họ quảng cáo nhắm đối tượng (targeted advertising).

Một chiến lược quảng cáo phổ biến là chiến lược đeo bám quảng cáo (retargeting), khi các trang web theo dõi những trang nào bạn đã truy cập và sau đó hiển thị cho bạn các quảng cáo dựa trên dữ liệu này. Đó là lý do tại sao bạn thường thấy quảng cáo cho các sản phẩm mà bạn đã xem gần đây khi duyệt web.

Quảng cáo cũng được hiển thị dựa trên dữ liệu được thu thập về lịch sử tìm kiếm, vị trí, sở thích của bạn và thời gian trong ngày bạn truy cập một trang web.

Mặc dù quảng cáo được nhắm đối tượng và chia sẻ dữ liệu giữa các công ty là điều mà nhiều dùng không muốn xảy ra, nhưng một số trang web dựa vào thu nhập mà quảng cáo nhắm mục tiêu mang lại để có thể cung cấp dịch vụ miễn phí cho người dùng. [6]

* + 1. **Usability testing**

Usability Testing (kiểm tra khả năng sử dụng) là một quá trình trong đó người dùng thật dùng thử một sản phẩm, chẳng hạn như một ứng dụng hoặc trang web, để đánh giá điểm mạnh và điểm yếu của nó. Trong một bài tập Usability Testing điển hình, người dùng sẽ được yêu cầu điều hướng một trang web hoặc hoàn thành một số tác vụ trên một ứng dụng. Dựa trên phản hồi của bạn - những thách thức bạn đã trải qua, những thứ khiến bạn bối rối hoặc vấn đề về các tính năng - sản phẩm cuối cùng có thể được điều chỉnh và cải thiện cho phù hợp.

Các nhà nghiên cứu sử dụng công nghệ web tracking để tiến hành kiểm tra khả năng sử dụng từ xa liên tục, thu được hàng tấn dữ liệu có giá trị mà họ có thể sử dụng để điều chỉnh sản phẩm theo sở thích của khách hàng thực tế.

* 1. **Các vấn đề về pháp lý, tính hợp pháp**

Các quy định về thông tin cá nhân, quyền riêng tư và các điều kiện để thu thập, xử lý Dữ liệu cá nhân hợp pháp là những yếu tố ảnh hưởng rất lớn đến việc cách thức thu thập, lưu trữ, phân tích dữ liệu người dùng của các doanh nghiệp. Chúng tôi sẽ liệt kê 1 số quy định nổi bật mà các nhà phát triển web tracking có thể gặp phải.

### 1.4.1 Quy định chung về bảo vệ dữ liệu - General Data Protection Regulation (GDPR)

Ở Châu âu, vào năm 2018, Liên minh Châu Âu ban hành Quy định chung về bảo vệ dữ liệu - General Data Protection Regulation (GDPR), cho phép người dùng internet kiểm soát nhiều hơn cách dữ liệu của họ được thu thập và chia sẻ. GDPR định nghĩa dữ liệu cá nhân là bất kỳ phần thông tin nào liên quan đến một người có thể nhận dạng được. Theo GDPR, các trang web có khách truy cập ở Liên minh Châu Âu phải đưa ra rõ ​​ràng các phương pháp theo dõi web của họ và đảm bảo rằng dữ liệu họ chia sẻ định danh cá nhân. Ngoài ra, người dùng Internet ở Liên minh Châu Âu phải đồng ý trước khi các trang web có thể đặt cookie theo dõi trên trình duyệt của họ [2].

### 1.4.2 Đạo luật Quyền riêng tư của Người tiêu dùng California - California Consumer Privacy Act (CCPA)

Ở Mỹ, từ ngày 01/01/2020, Đạo luật Quyền riêng tư của Người tiêu dùng California - California Consumer Privacy Act (CCPA) đưa ra các quyền riêng tư về dữ liệu mới cho cư dân California - buộc các công ty hoạt động kinh doanh ở bang California phải thực hiện các thay đổi cấu trúc đối với các chính sách quyền riêng tư của họ. Luật này là một phản ứng đối với vai trò ngày càng tăng của dữ liệu cá nhân trong thực tiễn kinh doanh và các tác động về quyền riêng tư của cá nhân xung quanh việc thu thập, sử dụng và bảo vệ thông tin cá nhân. Nếu không tuân thủ CCPA có thể bị phạt tới $ 7,500 (USD) cho mỗi lần vi phạm [9].

### 1.4.3 Các văn bản pháp luật quy định về bảo vệ dữ liệu, thông tin cá nhân và quyền riêng tư tại Việt Nam

Ở Việt Nam, có khoảng 20 văn bản pháp luật liên quan về bảo vệ dữ liệu, thông tin cá nhân và quyền riêng tư. Trong đó có các văn bản, quy định nổi bật như Luật An toàn thông tin mạng 2015, Luật An ninh mạng 2018, …

Ths. Nguyễn Hương Ly - Cục Quản lý mật mã dân sự và Kiểm định sản phẩm mật mã có bài viết phân tích, so sánh rất chi tiết về các quy định pháp luật về bảo vệ dữ liệu, thông tin cá nhân và quyền riêng tư ở Việt Nam so với GDPR [10]. Dưới đây là 7 cấu phần chính nổi bật mà tác giả đề ra:

*1. Về khái niệm, định nghĩa thông tin cá nhân, quyền riêng tư*

2. *Các điều kiện để thu thập, xử lý dữ liệu cá nhân hợp pháp*

3. *Quyền của chủ thể dữ liệu và các trường hợp hạn chế quyền*

4. *Trách nhiệm, nghĩa vụ của bên thu thập, xử lý dữ liệu*

5. *Thiết chế giám sát độc lập của cơ quan chính quyền và sự phối hợp giữa các cơ quan có liên quan*

6. *Chuyển tải dữ liệu qua biên giới*

7. *Chế tài và xử phạt*

Chúng tôi khuyến khích những cá nhân, tổ chức làm các công việc liên quan đến các hoạt động thu thập, lưu trữ, phân tích dữ liệu người dùng nên đọc bài viết này và các văn bản pháp luật liên quan. Vì sao phải đọc kỹ các văn bản pháp luật? Phía dưới đây là 1 ví dụ

“Điều 17. Thu thập và sử dụng thông tin cá nhân, Luật An toàn thông tin mạng 2015 có quy định sau:

Tổ chức, cá nhân xử lý thông tin cá nhân có trách nhiệm sau đây:

a) Tiến hành thu thập thông tin cá nhân sau khi có sự đồng ý của chủ thể thông tin cá nhân về phạm vi, mục đích của việc thu thập và sử dụng thông tin đó.

b) Chỉ sử dụng thông tin cá nhân đã thu thập vào mục đích khác mục đích ban đầu sau khi có sự đồng ý của chủ thể thông tin cá nhân.

c) Không được cung cấp, chia sẻ, phát tán thông tin cá nhân mà mình đã thu thập, tiếp cận, kiểm soát cho bên thứ ba, trừ trường hợp có sự đồng ý của chủ thể thông tin cá nhân đó hoặc theo yêu cầu của cơ quan nhà nước có thẩm quyền”.

Tác giả có nhận xét như sau: “cách quy định chung chung và thiếu cụ thể trong Điều 17 nói trên có vẻ nhấn mạnh bảo vệ quyền tối thượng của chủ thể dữ liệu, tuy nhiên sẽ phi thực tế bởi một mặt tạo nhiều rủi ro pháp lý không lường trước cho bên xử lý dữ liệu (ví dụ các doanh nghiệp), mặt khác cũng tạo sự lạm dụng một khi người dùng chỉ cần ký xác nhận đồng ý ban đầu khi cung cấp dữ liệu (vốn thường dễ dàng có được) thì sẽ được coi là đủ điều kiện cho bất kỳ việc xử lý nào tiếp sau đó”.

# CHƯƠNG 2 TỔNG QUAN VỀ HỆ THỐNG STREAMING

## 2.1 Hệ thống streaming

### 2.1.1 Hệ thống real-time

Hệ thống real-time được chia làm 3 loại:

**Bảng 1: Phân loại hệ thống Real-time** [11]

|  |  |  |  |
| --- | --- | --- | --- |
| Phân loại | Ví dụ | Độ trễ được đo bằng | Sự dễ dãi cho vượt quá độ trễ. |
| Hard | Máy tạo nhịp tim, chống bó cứng phanh | Microseconds–milliseconds | Không cho phép - toàn bộ hệ thống bị bị sụp đổ, nguy hiểm đến tính mạng. |
| Soft | Hệ thống đặt vé máy bay, báo giá chứng khóan trực tuyến, VoIP (Skype) | Milliseconds–seconds | Thấp - không có lỗi hệ thống, không có nguy hiểm đến tính mạng. |
| Near | Skype video, home automation | Seconds–minutes | Cao - không có lỗi hệ thống, không gặp rủi ro. |

### 2.1.2 Hệ thống streaming là gì?

Đây được gọi là hệ thống dữ liệu trực tuyến (streaming data system) làm cho dữ liệu của nó có sẵn tại thời điểm ứng dụng khách cần. Hệ thống này không phải hard real-time, soft Real-time cũng không phải Near Real-time – hệ thống này là hệ thống phát trực tuyến (streaming).

Ví dụ: A theo dõi các tweet Twitter của B, B đăng tweet ngay khi A vừa lướt hết phiên và reload lại phiên, A có thể xem được các tweet vừa được đăng của B. Tuy nhiên trong thực tế, thời điểm A xem được tweet của B không hẳn là ngay lập tức, có thể là một khoảng thời gian rất lâu sau, A mở điện thoại lên mới có thể thấy tweet, hoặc A đang nhắn tin với một người dùng khác, hoặc A đang đọc1 tweet khác và chưa reload phiên bản, …

Với những trường hợp trên, thuật ngữ real-time chưa phù hợp với hệ thống theo dõi các tweet của người dùng mà streaming sẽ phù hợp hơn.

### 2.1.3 Lý do đề cập đến hệ thống streaming?

Web tracking và streaming, hai khái niệm tưởng chừng chẳng liên quan với nhau, nhưng thật sự 2 khái niệm này trong thực tế có sự tương hỗ rất lớn.

Lấy ví dụ về hệ thống streaming theo dõi các tweet của người dùng do Twitter thiết lập, theo thống kê của internet live stats [11], người dùng Twitter tạo ra khoảng 500 triệu tweets mỗi ngày, trung bình mỗi giây có 6,000 tweet được tạo ra. Việc xử lý 1 lượng lớn dữ liệu trong một khoảng thời gian như vậy tương tự như việc thu thập, phân tích, xử lý hàng ngàn, hàng triệu sự kiện được tạo ra bởi người dùng của một sàn thương mại điện tử.

### 2.1.4 Kiến trúc 1 hệ thống streaming:

Lấy ví dụ hệ thống streaming của Twitter ở trên, kiến trúc dưới đây sẽ mô tả Pipeline dữ liệu được xử lý:

Diagram

Description automatically generated

**Hình 1: Hệ thống kiến trúc Streaming** [11]

Browser, device—Các thiết bị tạo ra dữ liệu như app Twitter trên điện thoại, laptop, desktop.

**Collection tier**—Khi 1 người dùng đăng 1 tweet, nó được thu thập bởi dịch vụ của Twitter.

**Message queuing tier** —Twitter vận hành các trung tâm dữ liệu ở các vị trí trên toàn cầu và lớp Message queuing tier này đảm bảo cho các message không bị mất trong quá trình truyền tải dữ liệu, đảm bảo message gửi đến lớp analysis đẻ xử lý.

**Analysis tier**—Khi được gửi đến lớp Analysis, hệ thống sẽ xác định những người theo dõi nội dung tweet để họ có thể nhận được thông báo tweet vừa được đăng.

**Long-term storage tier**—Sau khi xử lý, tweet sẽ được lưu trữ trong các kho lưu trữ để người dùng có thể xem lại tweet.

**In-memory data store tier**—Các tweet tồn tại trong thời gian rất ngắn trong lớp lưu trữ memory.

**Data access**—Sau khi xử lý dữ liệu và lưu trữ trong bộ nhớ memory, lớp Data access chịu trách nhiệm gửi các tweet lên các thiết bị đầu cuối của người dùng như app Twitter trên Điện thoại, Laptop, Desktop.

Tùy vào rất nhiều yếu tố hoàn cảnh, lượng dữ liệu Streaming cần xử lý, khả năng tài chính của doanh nghiệp mà các lớp trên có thể được tách riêng, gộp chung hay giản bớt. Vd dữ liệu tracking có thể gửi từ các End Devices xuống thẳng lớp Message queue Trong tiểu luận này, Chúng tôi sẽ đề cập sâu vào 2 phần chính yếu của hệ thống Streaming là Message queue và Spark Streaming.

## 2.2 Message queue Tier

### 2.2.1 Message queue

#### 2.2.1.1 Message queue là gì?

Message queue có thể hiểu message queue là một hộp thư, cho phép các thành phần/service trong một hệ thống (hoặc nhiều hệ thống), gửi thông tin cho nhau [13]. Message queue hoạt động theo nguyên tắc hàng đợi vì nó thực hiện việc lấy message theo cơ chế FIFO – First In First Out.

#### 2.2.1.2 Thành phần của một message queue

Lấy ví dụ về hệ thống streaming theo dõi các tweet của người dùng. Một hệ thống sử dụng message queue thường có những thành phần sau đây:

Graphical user interface

Description automatically generated with low confidence

**Hình 2: Thành phần của một Message queue** [13]

**Producer:** Chương trình/service tạo ra thông tin, đưa thông tin vào message queue. Trong trường hợp này là các app Twitter của người dùng trên mobile, laptop, desktop tạo ra các tweet và gửi xuống hàng đợi.

**Message**: Thông tin được gửi đi (có thể là text, binary hoặc JSON). Trong trường hợp này, các message là các tweet của người dùng.

**Message queue**: Nơi chứa những message này, cho phép producer và consumer có thể trao đổi với nhau. Các tweet được đưa vào hàng đợi để chờ được xử lý theo cơ chế FIFO – First In First Out.

**Consumer:** Chương trình/service nhận message từ message queue và xử lý. Consumer trong trường hợp này là lớp phân tích sẽ nhận các tweet từ message queue và xử lý.

#### 2.2.1.3 Lợi ích của message queue trong web tracking

**Đảm bảo duration/recovery**: Vào giờ cao điểm, khi có quá nhiều dữ liệu được gửi tới hệ thống, message queue sẽ đảm bảo dữ liệu được lưu trữ đầy đủ trong hàng đợi, khi lớp phân tích đang xử lý nhưng bị crash hoặc lỗi, ta không lo bị mất dữ liệu; vì có thể lấy dữ liệu từ trong hàng đợi ra và thực thi lại.

**Dễ mở rộng hệ thống**: Vào giờ cao điểm, nhiều message được gửi vào hàng đợi, ta có thể tăng số lượng consumer/các máy chủ phân tích lên để xử lý được nhiều message hơn. Khi không cần ta có thể giảm lại.

**Phân tách hệ thống**: Giúp phân tách hệ thống thành nhiều phần nhỏ hơn, mỗi phần chỉ xử lý 1 chức năng nhất định. Thay vì thực hiện nhiều chức năng trong một tệp thực thi duy nhất, nhiều chương trình có thể trao đổi thông tin bằng cách gửi thông báo giữa các service, giúp chúng dễ dàng kiểm tra, gỡ lỗi, phát triển và mở rộng quy mô.

**Hỗ trợ rate limit, batching**: Trong nhiều trường hợp, năng lực xử lý hệ thống có hạn (chỉ có thể xử lý 300 message/s). Với message queue, ta có thể dần dần lấy mesage trong hàng đợi ra xử lý, không sợ thất lạc.

### 2.2.2 Apache Kafka

#### 2.2.2.1 Event streaming là gì?

Event streaming là hoạt động thu thập dữ liệu trong thời gian thực từ các nguồn sự kiện như cơ sở dữ liệu, cảm biến, thiết bị di động, dịch vụ đám mây và ứng dụng phần mềm dưới dạng luồng sự kiện, lưu trữ các luồng sự kiện này một cách lâu dài để truy xuất sau này, thao tác, xử lý và phản ứng với các luồng sự kiện trong thời gian thực cũng như hồi phục; và định tuyến các luồng sự kiện tới các công nghệ đích khác nhau nếu cần. Do đó, Event streaming đảm bảo một luồng dữ liệu liên tục để thông tin có mặt ở đúng nơi, vào đúng thời điểm.

#### 2.2.2.2 Apache Kafka là gì?

Apache Kafka là một distributed event streaming platform có khả năng xử lý hàng nghìn tỷ sự kiện/message/dữ liệu mỗi ngày. Ban đầu được thiết lập như một Message queue. Kể từ khi được tạo ra và cung cấp nguồn mở bởi LinkedIn vào năm 2011, Kafka đã nhanh chóng phát triển từ message queue thành một distributed event streaming platform chính thức.

#### 2.2.2.3 Cách thức hoạt động.

Kafka được xây dựng dựa trên mô hình publish/subscribe, tương tự như bất kỳ hệ thống message nào khác. Các ứng dụng (đóng vai trò là producer) gửi các messages (records) tới một Kafka node (broker) và những messages này sẽ được xử lý bởi các ứng dụng khác gọi là consumers. Các messages được gửi tới Kafka Node sẽ được lưu trữ trong một nơi gọi là topic và sau đó consumer có thể subscribe tới topic đó và lắng nghe những messages này. Messages có thể là bất cứ thông tin gì như giá trị cảm biến, hành động người dùng, … Các topic này sẽ được phân vùng (partitioned), tức là Topic sẽ được phân thành nhiều vùng trên Broker nhằm hỗ trợ cho việc xử lý song song [15].

Diagram

Description automatically generated

**Hình 3: Cách thức hoạt động của Broker** [15]

#### 2.2.2.4 Ưu điểm của Kafka

Diagram, schematic

Description automatically generated

**Hình 4: Ưu điểm của Apache Kafka** [16]

**HIGH THROUGHPUT** – Thông lượng cao

Gửi message với thông lượng giới hạn mạng bằng cách sử dụng một nhóm máy chủ phân tán có độ trễ thấp tới 2ms.

**SCALABLE** – Khả năng mở rộng

Các cụm phân tán có thể lên đến một nghìn nhà broker, hàng nghìn tỷ message mỗi ngày, hàng petabyte dữ liệu, hàng trăm nghìn phân vùng. Dễ dàng mở rộng hệ thống.

**PERMANENT STORAGE** – Lưu trữ message dài hạn

Lưu trữ các luồng dữ liệu một cách an toàn trong một cụm phân tán, bền bì, có khả năng chịu lỗi, khôi phục dữ liệu tốt, đảm bảo không bị mất dữ liệu.

**Reliability** – Độ tin cậy

Kafka được phân tán (distributed), phân vùng (partitioned), nhân bản (replicated) và chịu lỗi (fault tolerant), nên nó rất đáng tin cậy (reliable).

## 2.3 Analysis tier

Lớp phân tích bao gồm 1 hoặc nhiều máy chủ, service chịu trách nhiệm xử lý các message được gửi đến từ các luồng dữ liệu như Kafka. Trong bài tiểu luận này, nhóm quyết định sẽ nghiên cứu và sử dụng Spark Streaming để thiết lập lớp phân tích này.

### 2.3.1 Spark

Apache Spark là một open-source cluster computing framework viết bằng ngôn ngữ Scala và được phát triển sơ khởi vào năm 2009 bởi AMP Lab tại đại học California. Sau này, Spark đã được trao cho Apache Software Foundation vào năm 2013 và được phát triển cho đến nay. Nó cho phép xây dựng các mô hình dự đoán nhanh chóng với việc tính toán được thực hiện trên một nhóm các máy tính, có có thể tính toán cùng lúc trên toàn bộ tập dữ liệu mà không cần phải trích xuất mẫu tính toán thử nghiệm. Tốc độ xử lý của Spark có được do việc tính toán được thực hiện cùng lúc trên nhiều máy khác nhau. Đồng thời việc tính toán được thực hiện ở bộ nhớ trong (in-memories) hay thực hiện hoàn toàn trên RAM.

#### 2.3.1.1 Hoàn cảnh ra đời

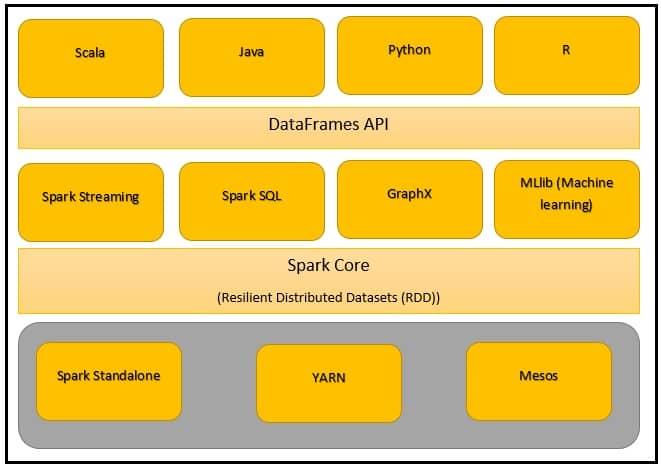
Matei Zaharia, cha đẻ của Spark, sử dụng Hadoop từ những ngày đầu. Đến năm 2009 ông viết Apache Spark để giải quyết những bài toán học máy ở đại học UC Berkely vì Hadoop MapReduce hoạt động không hiệu quả cho những bài toán này. Rất sớm sau đó ông nhận ra rằng Spark không chỉ hữu ích cho học máy mà còn cho cả việc xử lý luồng dữ liệu hoàn chỉnh.

Diagram

Description automatically generated

**Hình 5: Hệ thống Apache Spark** [17]

#### 2.3.1.2 Kiến trúc Spark



**Hình 6: Kiến trúc Spark** [17]

**Cluster Managers**: Spark có thể chạy trên nhiều loại Cluster Managers như Hadoop YARN, Apache Mesos hoặc trên chính cluster manager được cung cấp bởi Spark được gọi là Standalone Scheduler.

**Apache Spark Core – Resilient Distributed Datasets (RDD):** cung cấp những chức năng cơ bản nhất của Spark như lập lịch cho các tác vụ, quản lý bộ nhớ, fault recovery, tương tác với các hệ thống lưu trữ.

**Main Engine**: Spark SQL (SQL and HiveQL), Spark Streaming, machine learning and graph processing:

* **Spark SQL**: cho phép truy vấn dữ liệu cấu trúc qua các câu lệnh SQL. Spark SQL có thể thao tác với nhiều nguồn dữ liệu như Hive tables, Parquet, và JSON.
* **Spark Streaming**: cung cấp API để dễ dàng xử lý dữ liệu stream.
* **MLlib**: Cung cấp rất nhiều thuật toán của học máy như: classification, regression, clustering, collaborative filtering…
* **GraphX**: là thư viện để xử lý đồ thị.

**DataFrames API**: Cung cấp API Dataframe cho các ngôn ngữ lập trình Java, Scala, Python, R.

**Java**, **Scala**, **Python***,* **R**: Các ngôn ngữ lập trình có thể sử dụng Spark.

#### 2.3.1.3 Cơ chế hoạt động

Spark hoạt động dựa trên đặc tính file RDD.

Resilient Distributed Datasets (RDD) là một cấu trúc dữ liệu cơ bản của Spark. Nó là một tập hợp bất biến phân tán của một đối tượng. Mỗi dataset trong RDD được chia ra thành nhiều phần vùng logical. Có thể được tính toán trên các node khác nhau của một cụm máy chủ (cluster).

RDDs có thể chứa bất kỳ kiểu dữ liệu nào của Python, Java, hoặc đối tượng Scala, bao gồm các kiểu dữ liệu do người dùng định nghĩa. Thông thường, RDD chỉ cho phép đọc, phân mục tập hợp của các bản ghi. RDDs có thể được tạo ra qua điều khiển xác định trên dữ liệu trong bộ nhớ hoặc RDDs, RDD là một tập hợp có khả năng chịu lỗi mỗi thành phần có thể được tính toán song song.

Khi thực thi MapReduce, RDD lưu trữ trạng thái của bộ nhớ dưới dạng một đối tượng trên các công việc và đối tượng có thể chia sẻ giữa các công việc đó. Việc xử lý dữ liệu trong bộ nhớ nhanh hơn 10 đến 100 lần so với Network và Disk.

### 2.3.2 Spark Streaming

#### 2.3.2.1 Spark Streaming là gì?

Spark Streaming là một phần mở rộng của Spark API core cho phép các Data Engineers và Data Scientists xử lý dữ liệu thời gian thực từ nhiều nguồn khác nhau bao gồm (nhưng không giới hạn ở) Kafka, Flume, and Amazon Kinesis và có thể được xử lý bằng cách sử dụng các thuật toán phức tạp, thực thi các hàm như Map, Reduce, Join và Windows. Dữ liệu đã xử lý này có thể được đẩy ra file systems, Databases, hoặc Live Dashboards.



**Hình 7: Spark Streaming** [18]

#### 2.3.2.2 Cách hoạt động:

Spark Streaming nhận các luồng dữ liệu đầu vào trực tiếp và chia dữ liệu thành các lô, sau đó được xử lý bởi công cụ Spark để tạo ra luồng kết quả cuối cùng theo lô.



**Hình 8. Spark Streaming Pipeline** [18]**.**

Từ khóa quan trọng trong Spark Streaming thường được nhắc đến là discretized stream hay Dstream đại diện cho một luồng dữ liệu liên tục. DStream có thể được tạo từ các luồng dữ liệu đầu vào từ các nguồn như Kafka và Kinesis từ các DStream khác. Trong đó, một DStream được biểu diễn dưới dạng một chuỗi các RDD. Chính vì sử dụng các chuỗi RDD, cấu trúc dữ liệu quan trọng của Spark mà ta có thể áp dụng những thành phần khác của Spark như Spark MLlib để ứng dụng machine learning, hay Spark SQL để query dữ liệu.

Spark cho phép lập trình với ngôn ngữ Scala, Java và Python thông qua thư viện pyspark.

#### 2.3.2.3 Bốn ưu điểm chính của Spark Streaming

Phục hồi nhanh chóng từ các sự cố và lỗi.

Cân bằng tải và sử dụng tài nguyên tốt hơn.

Kết hợp Streaming data với datasets và các truy vấn tương tác.

Tích hợp với các thư viện xử lý nâng cao (SQL, machine learning, graph processing).

# CHƯƠNG 3. TỔNG QUAN VỀ MERN STACK

MERN Stack là một bộ mã nguồn mở giúp xây dựng một trang web hoàn chỉnh, các công nghệ đều có điểm chung là đều liên quan đến Javascript cũng đang hot hiện nay. Nó bao gồm các công nghệ như: MongoDB, ExpressJS, ReactJS, NodeJS.

Diagram

Description automatically generated with medium confidence

**Hình 9: Kiến trúc của MERN STACK** [19]

## React.JS

Lớp trên cùng của MERN Stack là React.JS, framework JavaScript khai báo để tạo các ứng dụng phía client trong HTML. React cho phép bạn xây dựng các giao diện phức tạp thông qua các Thành phần đơn giản, kết nối chúng với dữ liệu trên máy chủ phụ trợ của bạn và hiển thị chúng dưới dạng HTML.

Điểm mạnh của React là xử lý các giao diện theo hướng dữ liệu, trạng thái với code và pain đơn giản nhất, đồng thời nó có tất cả các bell và whistles mà bạn mong đợi từ một framework web hiện đại: hỗ trợ tuyệt vời cho forms, error handling, events, lists, … [19]

## Express.JS và Node.JS

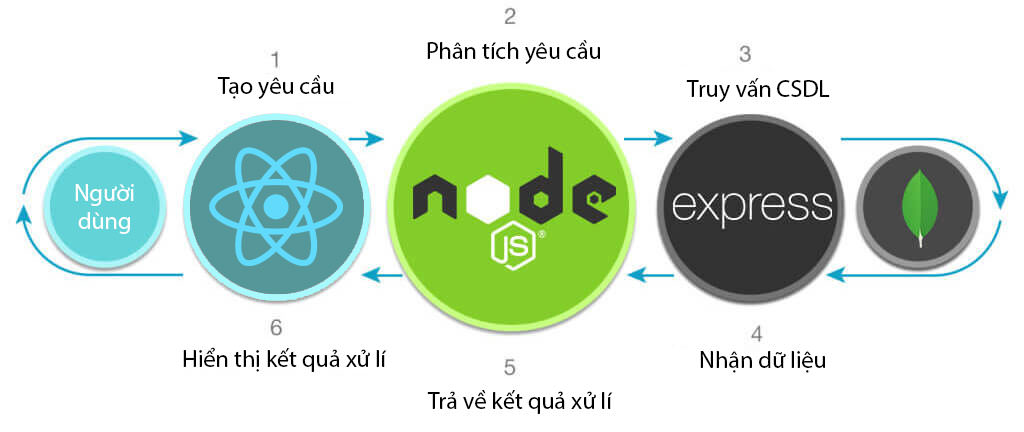
Lớp tiếp theo là framework Express.js, chạy bên trong Node.js server. Express.js tự nhận mình là một “framework web nhanh chóng, đơn giản, tối giản cho Node.js” và đó thực sự là chính xác những gì nó đang có. Express.js có các mô hình mạnh mẽ để định tuyến URL (khớp URL đến với chức năng máy chủ) và xử lý các yêu cầu và phản hồi HTTP.

Bằng cách thực hiện các request HTTP XML (XHR) hoặc GET hoặc POST từ giao diện người dùng React.js gửi tới Express.js để thực thi các hàm. Các chức năng đó lần lượt sử dụng trình điều khiển MongoDB’s Node.js, hoặc thông qua lệnh gọi lại để sử dụng promises, để truy cập và cập nhật dữ liệu trong cơ sở dữ liệu MongoDB [19].

## MongoDB

MongoDB là cơ sở dữ liệu không quan hệ có nghĩa các dữ liệu được lưu trữ dưới dạng JSON. Nó được tạo trong giao diện người dùng React.js của bạn có thể được gửi đến máy chủ của Express.js, nơi chúng có thể được xử lý và (giả sử chúng hợp lệ) được lưu trữ trực tiếp trong MongoDB để truy xuất sau này. Một lần nữa, nếu bạn đang xây dựng trên đám mây [19].

## Hoạt động của MERN Stack



**Hình 10: Quy trình hoạt động của MERN STACK** [20]

Người dùng sẽ tương với các thành phần React.js UI ở giao diện người dùng của ứng dụng trong trình duyệt. Một máy chủ Express.js viết bằng Node.js sẽ phục vụ cho phần back end. Bất kỳ tương tác nào tạo ra một yêu cầu thay đổi dữ liệu sẽ được gửi đến cho máy chủ Express.js. Sau đó, máy chủ thu dữ liệu từ cơ sở dữ liệu MongoDB (trong trường hợp được yêu cầu) và đưa lên giao diện người dùng của ứng dụng.

## Ưu điểm của MERN Stack

### Thiết kế use case

Với việc phát triển ứng dụng bằng MERN Stack, lập trình viên sẽ tiết kiệm được rất nhiều thời gian và công sức. Chỉ cần viết code bằng ngôn ngữ duy nhất là JavaScript cho cả máy khách và máy chủ [20].

### 3.5.2 Có thể code Isomorphic

Chuyển code sang một framework khác được viết bởi một framework cụ thể sẽ dễ dàng hơn với sự hỗ trợ của MERN Stack [21].

### 3.5.3 Tính linh hoạt cao

Với MERN, Có thể thực hiện việc kiểm tra ứng dụng trên nền tảng đám mây một cách thuận lợi sau khi đã hoàn thành quy trình phát triển. Đồng thời, MERN cũng cho dễ dàng thêm thông tin bổ sung bằng cách thêm các trường vào biểu mẫu [21].

### 3.5.4 Tối ưu chi phí

Sử dụng MERN Stack để phát triển ứng dụng chỉ đòi hỏi các lập trình viên phải thông thạo JavaScript. Trong khi đó, việc dùng LAMP Stack lại cần nhiều hơn thế. Điều này giúp tiết kiệm chi phí thuê và đào tạo nhân công cho các hãng sản xuất công nghệ [21].

### 3.5.5 Có tốc độ cao và tái sử dụng được

Node.js là có tốc độ xử lý cực nhanh và là một framework có cấu trúc non-blocking. Trong khi đó, React.js là một JavaScript framework mã nguồn mở cung cấp khả năng bảo trì, kiểm tra và tái sử dụng [21].

### 3.5.6 Mã nguồn mở và có khả năng tương thích với đám mây

Toàn bộ công nghệ MERN Stack đều là mã nguồn mở có sẵn hoàn toàn miễn phí. Vì thế, việc sử dụng MERN Stack cho phép lập trình viên có nhiều sự lựa chọn với thư viện và các kho lưu trữ công cộng. Nó cũng cho phép giảm chi phí phát triển phần mềm hiệu quả. MongoDB hỗ trợ việc triển khai các chức năng đám mây trong ứng dụng bằng cách giảm chi phí của không gian đĩa [21].

# CHƯƠNG 4. Thiết kế, cài đặt và kiểm thử

## 4.1 Sơ đồ hoạt động

Diagram

Description automatically generated

**Hình 11: Sơ đồ hoạt động**

**Mô tả vấn đề:**

Một trang web bán điện thoại có nhu cầu theo dõi các lượt click của người dùng, xem họ click vào sản phẩm nào và đề xuất những chiếc điện thoại có cấu hình tương tự.

**Mô tả cách thực hiện:**

Sử dụng Spark Streaming và các thư viện của Python, nhóm sẽ tải toàn bộ sản phẩm của cửa hàng, xử lý, chuẩn hoá dữ liệu và thiết lập 1 hàm đề xuất các sản phẩm tương tự dựa trên toàn bộ dữ liệu điện thoại của cửa hàng bằng tính khoảng cách Euclidean.

Khi người dùng click vào 1 sản phẩm bất kỳ trên web, id của sản phẩm sẽ được gửi vào Kafka, sau đó Spark Streaming sẽ lắng nghe từng id sản phẩm được gửi tới, và thực thi hàm đề xuất. Sau khi ra được những sản phẩm tương tự nhất. Spark Streaming sẽ gửi các sản phẩm tương tự này vào MongoDB. Client sẽ request và lấy dữ liệu và hiển thị lên trang web.

## 4.2 Môi trường cài đặt và các công nghệ liên quan

* + 1. **Môi trường cài đặt web**
* Sử dụng NodeJS phiên bản v16.13.1
* Sử dụng trình quản lý các thư viện javascript npm phiên bản 8.1.2
* Sử dụng cơ sở dữ liệu online MongoDB Atlas.

Để cài đặt Node.JS và Npm thì ta download tại đây <https://nodejs.org/en/download/>

Text

Description automatically generated

**Hình 12: Phiên bản của node**

Text

Description automatically generated

**Hình 13: Phiên bản npm**

### 4.2.2 MongoDB

Graphical user interface, application, Word

Description automatically generated

**Hình 14: Giao diện MongoDB**

Đăng kí tài khoản mongoDB online sau đó thiết lập tên database và collections.

A screenshot of a computer

Description automatically generated

**Hình 15: Database và Collections**

**Database**: myFirstDatabase

**Collections**:

* Categories: Danh mục các hãng điện thoại
* Payments: Thanh toán sản phẩm
* Products: Danh sách tất cả các sản phẩm
* Similarities: Danh sách các \_id sản phẩm tương tự.
* Trackings: Danh sách các \_id người dùng và \_id sản phẩm khi người dùng click
* Users: Danh sách các tài khoản người dùng.

**Mô tả dữ liệu:**

* **Users**

**Bảng 2: Thuộc tính bảng Users**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| id | Mã người dùng | {type: ObjectId} |  |
| name | Tên người dùng | {type: String, required: true, trim: true} |  |
| email | Email | {type: String, required: true, unique: true, trim: true} |  |
| password | Mật khẩu | {type: String, required: true} |  |
| role | Quyền | {type: Number, default: 0} | Role name:   * 0: User * 1: Admin |
| cart | Giỏ hàng | {type: Array, default: []} |  |

* **Products**

**Bảng 3: Thuộc tính bảng Products**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| \_id | id | {type: ObjectId} |  |
| Product\_id | Mã sản phẩm | {type: String, unique: true, trim: true, required: true} |  |
| Title | Tiêu đề sản phẩm | {type: String, trim: true, required: true} |  |
| Price | Giá sản phẩm | {type: Number, required: true, unique: true, trim: true} |  |
| Description | Mô tả sản phẩm | {type: String, required: true} |  |
| Content | Nội dung của sản phẩm | {type: String, required: true} |  |
| Images | Hình ảnh của sản phẩm | {type: Object, required: true} |  |
| Category | Thể loại | {type: String, required: true} |  |
| Checked | Kiểm tra | {type: Boolean, default: false} |  |
| Sold | Số lượng đã bán | {type: Number, default: 0} |  |
| screenSize | Màn hình | {type: String, trim: true, required: true} |  |
| Ram | Ram của sản phẩm | {type: Number, trim: true, required: true} |  |
| Memory | Dung lượng bộ nhớ của sản phẩm | {type: Number, trim: true, required: true} |  |
| Camera | Camera | {type: String, required: true} |  |
| Pin | Dung lượng pin của sản phẩm | {type: Number, trim: true, required: true} |  |
| Status | Tình trạng sản phẩm | {type: String, required: true} |  |
| Color | Màu sắc sản phẩm | {type: String, required: true} |  |

* **Trackings**

**Bảng 4: Thuộc tính bảng Trackings**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| \_id | id | {type: ObjectId} |  |
| Id\_user | Mã người dùng | {type: String, required: true} |  |
| Id\_product | Mã sản phẩm | {type: String, required: true} |  |

* **Similarities**

**Bảng 5: Thuộc tính bảng Similarities**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| \_id | id | {type: ObjectId} |  |
| idProduct | Mã sản phẩm | {type: String, required: true} |  |
| listId | Danh sách các mã sản phẩm tương tự | {type: Array, required: true} |  |

* **Payments**

**Bảng 6: Thuộc tính bảng Payments**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| \_id | id | {type: ObjectId} |  |
| User\_id | Mã người dùng | {type: String, required: true} |  |
| name | Tên người dùng | {type: String, required: true} |  |
| email | Email | {type: String, required: true} |  |
| paymentId | Mã giao dịch | {type: String, required: true} |  |
| Address | Địa chỉ | {type: Object, required} |  |
| cart | Giỏ hàng | {type: Array, default: []} |  |
| Status | Tình trạng | {type: boolean, default: false} |  |

* **Categories**

**Bảng 7: Thuộc tính bảng Categories**

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Ý nghĩa | Kiểu dữ liệu | Ghi chú |
| \_id | id | {type: ObjectId} |  |
| name | Tên danh mục | {type: String, required: true, trim: true, unique: true} |  |

* + 1. **Cài đặt môi trường cho Apache Hadoop, Apache Spark, Apache Kafka và**
* **Apache Hadoop 2.7**

Graphical user interface, application

Description automatically generated

**Hình 16: Môi trường Hadoop**

* **Apache Spark 3.0.2**

Graphical user interface, text, application

Description automatically generated

**Hình 17: Môi trường Spark**

* **Apache Kafka 2.12-2.8.0**

Graphical user interface, application

Description automatically generated

**Hình 18: Môi trường Kafka**

* **Apache Zookeeper-3.7.0**

Graphical user interface, text, application

Description automatically generated

**Hình 19: Môi trường Zookeeper**

**Cài đặt môi trường path**

%HADOOP\_HOME%\bin

%SPARK\_HOME%\bin

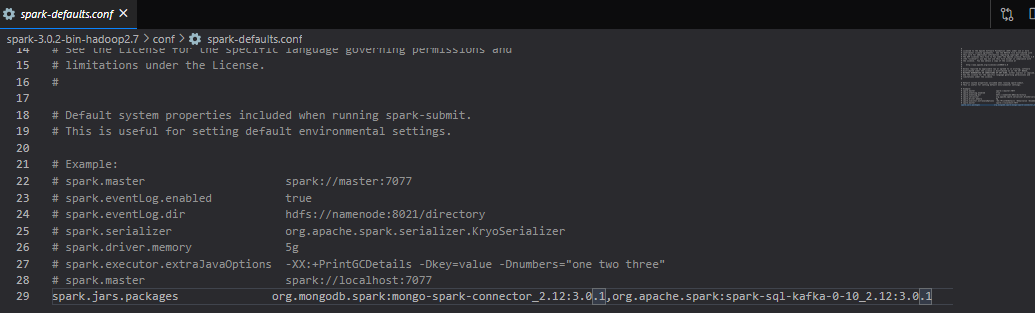
%ZOOKEEPER\_HOME%\bin

%KAFKA\_HOME%\bin

* **Java 1.8**
* **Các thư viện hỗ trợ:**

Thư viện hổ trợ kết nối mongodb: org.mongodb.spark:mongo-spark-connector\_2.12:3.0.1

Thư viện hỗ trợ query dữ liệu trên kafka Streaming data:org.apache.spark:spark-sql-kafka-0-10\_2.12:3.0.1.



**Hình 20: Thêm thư viện spark-default.conf**

## Thiết lập mô hình đề xuất các sản phẩm tương tự với Root mean square error.

* + 1. **Tiền xử lý dữ liệu**

Nhóm sẽ kết nối với mongodb để lấy dữ liệu products về và tiền xử lý dữ liệu. Đối với dữ liệu phân loại như category, status, color. Đưa chúng về dạng nhị phân. Thêm vào đó nhóm sử dụng StandardScaler chuẩn hóa dữ liệu. Để tính mức độ tương đồng trên sản phẩm.

A screenshot of a computer

Description automatically generated with medium confidence

**Hình 21: Dữ liệu sau khi tiền xử lý**

Text

Description automatically generated

**Hình 22: Dữ liệu sau khi StandardScaler**

* + 1. **Thiết lập hàm xử lý sản phẩm tương tự dựa trên Eclude**

Graphical user interface, text

Description automatically generated

**Hình 23: Hàm tính toán mức độ tương đồng**

Hàm này nhiệm vụ sẽ tính khoảng cách giữa sản phẩm click với tất cả các sản phẩm còn lại sau đó sắp xếp theo mức độ tương đồng nhiều nhất đến thấp nhất

## Quy trình hoạt động của hệ thống Web tracking:

Sau đây nhóm sẽ mô tả lại quy trình hoạt động của hệ thống Web tracking theo dõi hoạt động Click sản phẩm của người dùng và gợi ý những sản phẩm tương tự sản phẩm người dùng vừa Click theo thời gian thực.

**Quy trình:**

Sau khi đăng nhập trang web. Thì có giao diện như hình dưới.

Graphical user interface, application

Description automatically generated

**Hình 24: Giao diện web**

**Bước 1: Chọn sản phẩm muốn xem:**

Khi người dùng Click vào hình ảnh sản phẩm bất kì thì \_id của sản phẩm này sẽ truyền xuống back end (node.js). Từ đây, \_id sẽ được gửi xuống vào topic “clickcount” của lớp Kafka. Sau khi dữ liệu được truyền xuống thì Apache Kafka sẽ nhận lấy \_id này và bỏ vào hàng đợi.

Graphical user interface, application

Description automatically generated

**Hình 25: Chi tiết sản phẩm**

Text

Description automatically generated with medium confidence

**Hình 26: \_id gửi xuống nodejs**

**Bước 2**: Ngay Kafka nhận được 1 message, ngay lập tức, Spark Streaming trước đó đã được subscribe topic “clickcount” sẽ được trigger, lấy \_id sản phẩm và tính toán để đưa ra 10 sản phẩm tương tự như hình dưới.

Table

Description automatically generated

**Hình 27: Danh sách sản phẩm gợi ý**

**Bước 3**: Sau khi đưa ra được các sản phẩm tương tự, hệ thống sẽ lưu trữ 5 id sản phẩm vào bộ nhớ dài hạn MongoDB.



**Hình 28: Danh sách mã sản phẩm**

**Bước 4**: Khi đó Front end sẽ lấy dữ liệu và hiển thị trên web sau 3s Click.

Graphical user interface, text, application, Word

Description automatically generated

**Hình 29: Gợi ý 5 sản phẩm**

# PHẦN 3: PHẦN KẾT LUẬN

## Kết quả đạt được

Xây dựng thành công hệ thống Web tracking theo dõi hoạt động Click sản phẩm của người dùng và gợi ý những sản phẩm tương tự sản phẩm người dùng vừa Click theo thời gian thực.

## Hạn chế

* Kỹ thuật gợi ý sản phẩm chưa được tối ưu với từng người dùng.
* Thời gian hiển thị đề xuất chưa được tối ưu độ trễ.

## Hướng phát triển

* Tối ưu kỹ thuật gợi ý sản phẩm dựa trên việc phân tích thói quen sở thích hành vi của người dùng, vd: số lượng click người dùng vào 1 sản phẩm, đánh giá của người dùng khi mua sản phẩm, bình luận đánh giá của người dùng về sản phẩm sau khi mua, … Nghiên cứu sử dụng các thuật toán đề xuất Content-based, Neighborhood-Based Collaborative Filtering.
* Tối ưu thời gian đề xuất sản phẩm.

TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Thương ban chỉ đạo bộ công an, "Phát triển thương mại điện tử trở thành một trong những lĩnh vực tiên phong của nền kinh tế số," 02 03 2021. [Online]. Available: https://moit.gov.vn/bao-ve-nen-tang-tu-tuong-cua-dang/phat-trien-thuong-mai-dien-tu-tro-thanh-mot-trong-nhung-linh.html. |
| [2] | Avast, "A Complete Guide to Web tracking (and How to Avoid It)," 06 12 2021. [Online]. Available: https://www.avast.com/c-web-tracking#gref . |
| [3] | Reflectiz, "The Cybersecurity Effects of Fourth-Parties on Websites," 09 08 2020. [Online]. Available: https://www.reflectiz.com/blog/the-cybersecurity-effects-of-fourth-parties-on-websites-2/. |
| [4] | Hotjar, "Website tracking guide," 22 09 2021. [Online]. Available: https://www.hotjar.com/website-tracking/ . |
| [5] | N. Schmucker, "Web tracking," 2011. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.474.8976&rep=rep1&type=pdf . |
| [6] | T. Urban, "Beyond the Front Page: Measuring Third Party Dynamics in the Field," 03 02 2021. [Online]. Available: https://arxiv.org/abs/2001.10248 . |
| [7] | CookiePro, "Website Tracking Technologies," 17 09 2021. [Online]. Available: https://www.cookiepro.com/knowledge/website-tracking-technologies/#sec6. |
| [8] | Buitwith, "Google Analytics Usage Statistics," [Online]. Available: https://trends.builtwith.com/analytics/Google-Analytics . |
| [9] | CookiePro, "CCPA Compliance," 01 01 2020. [Online]. Available: https://www.cookiepro.com/regulations/ccpa/. |
| [10] | Ths. Nguyễn Hương Ly Cục QLMMDS & KĐSPMM, "Pháp luật hiện hành của Việt Nam về bảo vệ dữ liệu, thông tin cá nhân và quyền riêng tư," 25 12 2020. [Online]. Available: https://nacis.gov.vn/nghien-cuu-trao-doi/-/view-content/214123/phap-luat-hien-hanh-cua-viet-nam-ve-bao-ve-du-lieu-thong-tin-ca-nhan-va-quyen-rieng-tu. |
| [11] | A. G. Psaltis, Streaming Data: Understanding the real-time pipeline 1st Edition, Manning, 2017. |
| [12] | Twitter, "Twitter Usage Statistics," 2022. [Online]. Available: https://www.internetlivestats.com/twitter-statistics/. |
| [13] | P. H. HOÀNG, "MESSAGE QUEUE – BỘ PHẬN KHÔNG THỂ THIẾU TRONG CÁC HỆ THỐNG LỚN VÀ MICROSERVICE ARCHITECTURE," 08 10 2019. [Online]. Available: https://toidicodedao.com/2019/10/08/message-queue-la-gi-ung-dung-microservice/. |
| [14] | AWS, "Message Queues," [Online]. Available: https://aws.amazon.com/vi/message-queue/. |
| [15] | Dat Bui, "005: Bao nhiêu partition là đủ cho một topic trong Apache Kafka," 25 08 2021. [Online]. Available: https://viblo.asia/p/005-bao-nhieu-partition-la-du-cho-mot-topic-trong-apache-kafka-V3m5WQxQZO7. |
| [16] | Đình Tài, "Tổng quan về Apache Kafka," 27 05 2020. [Online]. Available: https://viblo.asia/p/tong-quan-ve-apache-kafka-RnB5p3OwlPG. |
| [17] | Hoang Trong Hieu, "Tổng quan về Apache Spark cho hệ thống Big Data," 29 6 2017. [Online]. Available: https://viblo.asia/p/tong-quan-ve-apache-spark-cho-he-thong-big-data-RQqKLxR6K7z. |
| [18] | Spark, "Spark Streaming Programming Guide," 13 10 2021. [Online]. Available: https://spark.apache.org/docs/latest/streaming-programming-guide.html. |
| [19] | MongoDB, "MERN Stack Explained," 2021. [Online]. Available: https://www.mongodb.com/mern-stack. |
| [20] | Not Cuder, "Series Javascript thần thánh – Javascript từ một ngôn ngữ sida trở thành bá chủ như thế nào?," 21 4 2018. [Online]. Available: https://notcuder.com/series-javascript-than-thanh-javascript-tu-mot-ngon-ngu-sida-tro-thanh-ba-chu-nhu-the-nao/. |
| [21] | Geneva Clark, "7 Advantages of Developing Apps with MEAN Stack," 11 09 2017. [Online]. Available: https://www.zeolearn.com/magazine/7-advantages-of-developing-apps-with-mean-stack. |