BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**NGHIÊN CỨU VÀ XÂY DỰNG EXTENSION CHO TRÌNH DUYỆT GOOGLE CHROME, ỨNG DỤNG XÂY DỰNG KỊCH BẢN KIỂM THỬ PHẦN MỀM NỀN TẢNG WEB**

**Sinh viên thực hiện:**

**Nguyễn Văn Dũng Lớp CNTT 3, khoa Công nghệ thông tin**

Đặng Đức Bảo Dũng Lớp CNTT 6, khoa Công nghệ thông tin

Phạm Văn Cương Lớp CNTT 1, khoa Công nghệ thông tin

Người hướng dẫn: Ths. Nguyễn Xuân Hoàng

**Hà Nội, 5/2023**

BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**NGHIÊN CỨU VÀ XÂY DỰNG EXTENSION CHO TRÌNH DUYỆT GOOGLE CHROME, ỨNG DỤNG XÂY DỰNG KỊCH BẢN KIỂM THỬ PHẦN MỀM NỀN TẢNG WEB**

**Sinh viên thực hiện:**

**Nguyễn Văn Dũng. Giới tính: Nam. Dân tộc: Kinh**

**Lớp, khoa: Công nghệ thông tin 3, Công nghệ thông tin Năm thứ: 3/4**

**Ngành học: Công nghệ thông tin**

Đặng Đức Bảo Dũng. Giới tính: Nam. Dân tộc: Kinh

Lớp, khoa: Công nghệ thông tin 6, Công nghệ thông tinNăm thứ: 3/4

Ngành học: Công nghệ thông tin

Phạm Văn Cương. Giới tính: Nam. Dân tộc: Kinh

Lớp, khoa: Công nghệ thông tin 1, Công nghệ thông tin Năm thứ: 3/4

Ngành học: Công nghệ thông tin

Người hướng dẫn: Ths. Nguyễn Xuân Hoàng

**Hà Nội, 5/2023**

# MỤC LỤC

[MỤC LỤC 1](#_Toc135304184)

[DANH MỤC HÌNH ẢNH 2](#_Toc135304185)

[CHƯƠNG 1: KHẢO SÁT VÀ PHÂN TÍCH BÀI TOÁN KIỂM THỬ PHẦN MỀM 4](#_Toc135304186)

[1.1. Tổng quan về bài toán 4](#_Toc135304187)

[1.2.Định hướng và giải pháp 4](#_Toc135304188)

[1.3.Phạm vi của đề tài 4](#_Toc135304189)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 5](#_Toc135304190)

[2.1.Kiểm thử phần mềm 5](#_Toc135304192)

[2.1.1.Khái niệm 5](#_Toc135304193)

[2.1.2.Mục tiêu của kiểm thử phần mềm 5](#_Toc135304194)

[2.1.3.Tại sao cần phải kiểm thử phần mềm? 5](#_Toc135304195)

[2.1.4.Quy trình kiểm thử 6](#_Toc135304196)

[2.1.5.Các loại kiểm thử 6](#_Toc135304197)

[2.1.5.1.Kiểm thử chức năng 6](#_Toc135304198)

[2.1.5.2.Kiểm thử khả năng tương thích 7](#_Toc135304199)

[2.1.5.3.Kiểm thử khả năng sử dụng 8](#_Toc135304200)

[2.1.5.4.Kiểm thử bảo mật 9](#_Toc135304201)

[2.1.5.5.Kiểm thử hiệu suất 10](#_Toc135304202)

[2.1.5.6.Kiểm thử độ chịu tải 11](#_Toc135304203)

[2.1.6.Kĩ thuật kiểm thử phần mềm 12](#_Toc135304204)

[2.1.6.1.Kiểm thử hộp trắng (White box testing) 12](#_Toc135304205)

[2.1.6.2.Kiểm thử hộp đen (Black box testing) 13](#_Toc135304206)

[2.1.7.Các cấp độ kiểm thử phần mềm 14](#_Toc135304207)

[2.2.Ngôn ngữ Javascipt 17](#_Toc135304208)

[2.3.Extension 19](#_Toc135304209)

[2.3.1.Khái niệm 19](#_Toc135304210)

[2.3.2.Kiến trúc của extension 20](#_Toc135304211)

[2.3.3.Ngôn ngữ lập trình xây dựng extension 21](#_Toc135304212)

[2.3.4.Cách tìm và cài đặt extension trên trình duyệt 22](#_Toc135304213)

[2.3.5.Cách gỡ, xóa bỏ tiện ích mở rộng trong Google Chrome 24](#_Toc135304214)

[CHƯƠNG 3: THỰC NGHIỆM 27](#_Toc135304215)

[3.1.Kế hoạch thực nghiệm 27](#_Toc135304216)

[3.1.1.Kế hoạch 27](#_Toc135304217)

[3.1.2.Mục đích 27](#_Toc135304218)

[3.1.3. Cách cài extension vào trình duyệt 28](#_Toc135304219)

[3.2. Thực nghiệm 30](#_Toc135304220)

[3.3.Giải thích hoạt động 32](#_Toc135304221)

[CHƯƠNG 4: XÂY DỰNG EXTENSION CRAWL DATA TRONG ĐẠI HỌC ĐIỆN TỬ ĐẠI HỌC CÔNG NGHIỆP 40](#_Toc135304222)

[4.1. Mục đích 40](#_Toc135304223)

[4.2.Cài đặt ứng dụng 40](#_Toc135304224)

[4.3.Giải thích hoạt động 42](#_Toc135304225)

[KẾT LUẬN 47](#_Toc135304226)

[TÀI LIỆU THAM KHẢO 48](#_Toc135304227)

# DANH MỤC HÌNH ẢNH

[Hình ảnh 2.1: Quy trình kiểm thử 6](#_Toc135303999)

[Hình ảnh 2.2: Kiểm thử tương thích 8](#_Toc135304000)

[Hình ảnh 2.3: Hình ảnh minh họa về kiểm thử bảo mật 10](#_Toc135304001)

[Hình ảnh 2.4: Hình ảnh minh họa về kiểm thử hiệu suất 11](#_Toc135304002)

[Hình ảnh 2.5: Hình ảnh minh họa về kiểm thử độ chịu tải 12](#_Toc135304003)

[Hình ảnh 2.6: Kiểm thử hộp trắng 13](#_Toc135304004)

[Hình ảnh 2.7: Kiểm thử hộp đen 14](#_Toc135304005)

[Hình ảnh 2.8: Các cấp độ kiểm thử phần mềm 15](#_Toc135304006)

[Hình ảnh 2.9: JavaScript logo 17](file:///C:\Users\minh0\Downloads\BC_NCKH_Nguyen_Van_Dung-3-1.docx#_Toc135304007)

[Hình ảnh 2.10: Kiến trúc của Extension 21](#_Toc135304008)

[Hình ảnh 2.11: Truy cập cửa hàng Chorme trực tuyến 23](#_Toc135304009)

[Hình ảnh 2.12: Truy cập cửa hàng Chorme trực tuyến 23](#_Toc135304010)

[Hình ảnh 2.13: Thêm tiện ích 24](#_Toc135304011)

[Hình ảnh 2.14: Hiển thị Extension 25](#_Toc135304012)

[Hình ảnh 2.15: Chọn tiện ích muốn xóa 25](#_Toc135304013)

[Hình ảnh 2.16: Xóa tiện ích 26](#_Toc135304014)

[Hình ảnh 2.17: Xác nhận xóa tiện ích 26](#_Toc135304015)

[Hình ảnh 3.1: Mở cài đặt trên chorme 28](#_Toc135304016)

[Hình ảnh 3.2: Chọn tiện ích 28](#_Toc135304017)

[Hình ảnh 3.3: Bật chế độ nhà phát triển 29](#_Toc135304018)

[Hình ảnh 3.4: Cho phép xử dụng Extension 29](#_Toc135304019)

[Hình ảnh 3.5: Ghim tiện ích mở rộng 30](#_Toc135304020)

[Hình ảnh 3.6: Tạo form tài khoản 30](#_Toc135304021)

[Hình ảnh 3.7: Chọn extension sử dụng 31](#_Toc135304022)

[Hình ảnh 3.8: Chuyển đổi dữ liệu qua google form 31](#_Toc135304023)

[Hình ảnh 3.9: Function sử lý sự kiện 32](#_Toc135304024)

[Hình ảnh 3.10: Cấu hình URL 32](#_Toc135304025)

[Hình ảnh 3.11: Trích xuất tab 33](#_Toc135304026)

[Hình ảnh 3.12: Thực thi script từ background 33](#_Toc135304027)

[Hình ảnh 3.13: Thực thi lắng nghe sự kiện 34](#_Toc135304028)

[Hình ảnh 3.14: Kiểm tra message 35](#_Toc135304029)

[Hình ảnh 3.15: Trả response cho content web 35](#_Toc135304030)

[Hình ảnh 3.16: Thực hiện quety đến trang đăng kí 36](#_Toc135304031)

[Hình ảnh 3.17: Thực thi API 36](#_Toc135304032)

[Hình ảnh 3.18: Thực thi TabId 37](#_Toc135304033)

[Hình ảnh 3.19: Thực thi query đến DOM 37](#_Toc135304034)

[Hình ảnh 3.20: Truyền message 38](#_Toc135304035)

[Hình ảnh 3.21: Kết quả thực thi 39](#_Toc135304036)

[Hình ảnh 4.1: Hệ thống hứng dữ liệu 41](#_Toc135304037)

[Hình ảnh 4.2: Cài đặt tiện ích craw**l** data 41](#_Toc135304038)

[Hình ảnh 4.3: Hệ thống đại học điện tử 42](#_Toc135304039)

[Hình ảnh 4.4: Kết quả craw**l** data 42](#_Toc135304040)

[Hình ảnh 4.5: Thực thi sự kiện 43](#_Toc135304041)

[Hình ảnh 4.6: Truyền dữ liệu 44](#_Toc135304042)

[Hình ảnh 4.7: Cấu hình chorm storage background 44](#_Toc135304043)

[Hình ảnh 4.8: Lắng nghe sự kiện 45](#_Toc135304044)

[Hình ảnh 4.9: Cấu hình manifest.json 46](#_Toc135304045)

MỞ ĐẦU

1. Đề tài và lý do lựa chọn đề tài

Phần mềm là một ngành công nghiệp phát triển nhanh chóng và có tính đa dạng cao. Tuy nhiên, để đảm bảo chất lượng sản phẩm phần mềm, việc kiểm thử là một yếu tố không thể thiếu. Trong quá trình kiểm thử phần mềm, việc xây dựng và sử dụng các kịch bản kiểm thử là cực kỳ quan trọng để đảm bảo sự chính xác, hiệu quả và độ tin cậy của sản phẩm.

Một trong những xu hướng phát triển của phần mềm hiện nay là ứng dụng web, và các trình duyệt web như Google Chrome đóng vai trò quan trọng trong quá trình này. Tuy nhiên, việc xây dựng và kiểm thử phần mềm web cũng đòi hỏi những công cụ hỗ trợ đặc biệt để đảm bảo hiệu quả và chất lượng sản phẩm. Trong bối cảnh đó, việc nghiên cứu và xây dựng extension cho trình duyệt Google Chrome, ứng dụng xây dựng kịch bản kiểm thử phần mềm nền tảng web trở thành một giải pháp tiềm năng để tối ưu hóa quá trình kiểm thử phần mềm web.

Extension cho trình duyệt Google Chrome là một phần mở rộng có thể tùy chỉnh và tích hợp vào trình duyệt, cung cấp cho người dùng các tính năng đa dạng, từ tăng cường bảo mật, quản lý mật khẩu đến kiểm tra chính tả. Với các tính năng linh hoạt và đa dạng này, extension cho trình duyệt Google Chrome có thể được sử dụng để xây dựng và kiểm thử phần mềm web một cách hiệu quả.

Ngoài ra, việc phát triển extension cho trình duyệt Google Chrome cũng sẽ giúp tăng cường khả năng tùy biến và mở rộng tính năng của trình duyệt, giúp người dùng có được trải nghiệm tốt hơn khi sử dụng trình duyệt.

Do đó, đề tài "Nghiên cứu và xây dựng extension cho trình duyệt Google Chrome, ứng dụng xây dựng kịch bản kiểm thử phần mềm nền tảng web" được lựa chọn nhằm tìm hiểu cách sử dụng extension cho trình duyệt Google Chrome trong việc xây dựng và kiểm thử phần mềm web.

1. Mục tiêu của đề tài

Hiểu được những khái niệm về kiểm thử, quy trình và cách thực hiện kiểm thử phần mềm. Tìm hiểu được lý thuyết, cách hoạt động của Extension từ đó ứng dụng để tạo một Extension kiểm thử phần mềm website của trường Đại học Công Nghiệp Hà Nội.

Nhằm thúc đẩy trào lưu nghiên cứu, phát triển các sản phẩm về Extension. Mở ra nhiều hướng nghiên cứu, ứng dụng vào kiểm thử nền tảng website.

1. Phương pháp nghiên cứu

Trước tiên nhóm nghiên cứu thực hiện tìm hiểu lý thuyết về extension, ngôn ngữ Javascript, kiểm thử phần mềm. Sau đó ứng dụng xây dựng một extension cơ bản để nắm vững kiến thức hơn.

Tiếp theo, nhóm tập trung vào việc nghiên cứu, phát triển bài toán kiểm thử website bằng trình duyệt extension.

1. Ý nghĩa khoa học và thực tiễn của đề tài

**Ý nghĩa khoa học:** Khai thác và phát triển các kỹ thuật mới trong lĩnh vực phần mềm và kiểm thử phần mềm. Nghiên cứu này sẽ giúp mở rộng hiểu biết về công nghệ phần mềm và cải thiện quy trình kiểm thử phần mềm nền tảng web. Đồng thời, nghiên cứu này cũng sẽ đóng góp vào sự phát triển của lý thuyết và ứng dụng trong lĩnh vực phần mềm và kiểm thử phần mềm.

**Ý nghĩa thực tiễn:** Giúp cải thiện quá trình kiểm thử phần mềm nền tảng web và tăng cường tính tùy biến và mở rộng tính năng của trình duyệt Google Chrome. Công cụ phát triển được trong đề tài sẽ giúp cho những người tham gia phát triển ứng dụng website có thể thực hiện kiểm thử phần mềm một cách dễ dàng và nhanh chóng hơn, đồng thời giúp cho người dùng có được trải nghiệm sử dụng trình duyệt tốt hơn. Ngoài ra, công cụ phát triển này cũng sẽ đóng góp vào sự phát triển của ngành công nghiệp phần mềm và giúp tăng cường sự cạnh tranh của các sản phẩm phần mềm.

1. Bố cục báo cáo

Quyển báo cáo nghiên cứu khoa học sẽ bao gồm các chương:

Chương 1: Khảo sát và phân tích bài toán kiểm thử phần mềm

Chương 2: Cơ sở lý thuyết

Chương 3: Thực nghiệm

Chương 4: Xây dựng Extension crawl data trong hệ thống đại học điện tử Đại học Công nghiệp Hà Nội

Phần kết luận

NỘI DUNG

## CHƯƠNG 1: KHẢO SÁT VÀ PHÂN TÍCH BÀI TOÁN KIỂM THỬ PHẦN MỀM

### Tổng quan về bài toán

Bài toán "Nghiên cứu và xây dựng extension cho trình duyệt Google Chrome, ứng dụng xây dựng kịch bản kiểm thử phần mềm nền tảng web" là một đề tài nghiên cứu x quan đến việc xây dựng một extension (tiện ích mở rộng) cho trình duyệt Google Chrome, giúp kiểm thử các ứng dụng web.

Extension là một chương trình được cài đặt trên trình duyệt, cho phép mở rộng tính năng của trình duyệt và cung cấp các chức năng bổ sung cho người dùng. Trong trường hợp này, extension được xây dựng nhằm hỗ trợ việc kiểm thử phần mềm nền tảng web.

Kịch bản kiểm thử phần mềm là một tập hợp các bước kiểm thử được thiết kế để đảm bảo rằng phần mềm hoạt động đúng như mong đợi và không có lỗi. Trong bài toán này, kịch bản kiểm thử được xây dựng để kiểm thử các ứng dụng web.

### Định hướng và giải pháp

Các chức năng chính của extension bao gồm tự động tương tác với các trang web, tạo ra các kịch bản kiểm thử tự động và thực hiện các kịch bản này để kiểm thử ứng dụng web. Extension cũng có thể cung cấp các báo cáo về kết quả kiểm thử và các lỗi được phát hiện trong quá trình kiểm thử.

### Phạm vi của đề tài

Phạm vi nghiên cứu và ứng dụng ở trang hệ thống website của nhà trường, đề tài có thể mở rộng phạm vi để áp dụng cho các trình duyệt web khác hoặc các nền tảng web khác ngoài phần mềm nền tảng web. Việc mở rộng phạm vi đề tài sẽ đòi hỏi nghiên cứu thêm về các công nghệ, tiêu chuẩn và kỹ thuật liên quan đến các nền tảng và trình duyệt khác.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT



### Kiểm thử phần mềm

#### Khái niệm

- Là quá trình kiểm tra sản phẩm phần mềm với mục đích:

* Đánh giá các sản phẩm công việc như yêu cầu, yêu cầu người dùng, thiết kế và code.
* Xác minh xem sản phẩm đó có thỏa mãn yêu cầu đặc tả hay không
* Xác thực xem sản phẩm đó có thỏa mãn những yêu cầu của người dùng không
* Xác định được các lỗi thường xuyên gặp
* Cung cấp thông tin đầy đủ cho các bên liên quan để cho phép họ đưa ra những quyết định, đặc biệt là về chất lượng của sản phẩm được kiểm thử
* Giảm mức độ rủi ro về chất lượng phần mềm

#### Mục tiêu của kiểm thử phần mềm

**Mục tiêu trực tiếp:**

* Xác định và phát hiện nhiều lỗi nhất có thể trong phần mềm được kiểm thử. Chứng minh rằng sản phẩm phần mềm phù hợp với đặc tả yêu cầu của nó
* Sau khi sửa chữa các lỗi đã xác định và kiểm tra lại, làm cho phần mềm đã được kiểm thử đến một mức độ chấp nhận được về chất lượng.
* Tạo các testcase chất lượng cao, thực hiện kiểm thử hiệu quả và tạo ra các báo cáo vấn đề đúng và hữu dụng.

**Mục tiêu gián tiếp:**

* Biên dịch một tài liệu về các lỗi phần mềm thường gặp nhằm mục đích ngăn ngừa và sửa chữa lỗi.

#### Tại sao cần phải kiểm thử phần mềm?

- Việc kiểm thử là cần thiết vì:

* Sản phẩm phần mềm nào cũng chắc chắn có lỗi: sản phẩm phần mềm là do con người xây dựng lên, dù con người có giỏi tới đâu cũng không thể đảm bảo rằng sản phẩm mình tạo ra là không có lỗi. Do đó sẽ cần một người, một nhóm hoặc một tổ chức độc lập kiểm thử xem sản phẩm đó có vấn đề, có lỗi hay không.
* Có thể giảm rủi ro có lỗi trong quá trình hoạt động, vận hành hoặc các lỗi tìm được khi đang được dùng trong thực tế, có thể sẽ rất tốn chi phí để khắc phục, sửa chữa thậm chí có thể gây ra nguy hiểm tới tính mạng con người.
* Việc kiểm thử giúp tìm hiểu độ tin cậy của phần mềm
* Tránh được sự kiện tụng của khách hàng: khách hàng mua sản phẩm, có lỗi hay vấn đề trên sản phẩm đó thì họ có thể kiện nhà sản xuất, nhà cung cấp sản phẩm đó

#### Quy trình kiểm thử

Quy trình kiểm thử phần mềm là chuỗi các hoạt động được tiến hành để thực hiện việc kiểm thử. Nó bao gồm hàng loạt các hoạt động được tiến hành theo phương pháp luận để giúp xác nhận sản phẩm phần mềm của bạn.



Hình ảnh 2.1: Quy trình kiểm thử

#### Các loại kiểm thử

##### Kiểm thử chức năng

* Functional Testing là bước kiểm thử cơ bản nhất cho bất kỳ một ứng dụng nào để đảm bảo rằng nó được làm việc theo đúng các yêu cầu, quy định đã đặt ra.
* Tương tự như các ứng dụng user-interface cơ bản khác, các ứng dụng di động cần đến sự tương tác giữa con người với chúng.

Kịch bản kiểm thử:

* Xác nhận rằng luôn có chuyến bay sẵn sàng được hiển thị cho một điểm đến nào đó và cho phép chọn ngày.
* Xác nhận ngày đã qua không được hiển thị trong kết quả tìm kiếm.
* Xác thực xem ứng dụng có hoạt động bình thường khi có cuộc gọi, tin nhắn, thông báo từ ứng dụng khác đến.
* Xác thực ứng dụng hỗ trợ mọi giao dịch cổng thanh toán như Visa, Mastercard, Paypal, vv theo yêu cầu của ứng dụng.
* Xác thực rằng người dùng nhận được thông báo lỗi thích hợp như lỗi Mạng Network.
* Xác thực rằng ứng dụng sẽ tiếp tục ở hoạt động cuối cùng trong trường hợp khởi động lại cứng hoặc sự cố hệ thống.

##### Kiểm thử khả năng tương thích

* Kiểm tra khả năng tương thích là bước quan trọng nhất khi nói đến kiểm thử ứng dụng di động.
* Mục đích của mỗi lần kiểm thử khả năng tương thích của ứng dụng di động là để đảm bảo các chức năng chính của ứng dụng hoạt động đúng như mong đợi trên một thiết bị cụ thể. Sự tương thích của ứng dụng chỉ mất một vài phút, và có thể lên kế hoạch cụ thể trước đó để có thể tiến hành trôi chảy nhất

Kịch bản kiểm thử:

* Xác minh rằng tìm kiếm chuyến bay được thực hiện thành công với thiết bị Android
* Xác minh rằng tìm kiếm chuyến bay được thực hiện thành công cho Apple iPad.

****

Hình ảnh 2.2: Kiểm thử tương thích

##### Kiểm thử khả năng sử dụng

Kiểm tra khả năng sử dụng đánh giá ứng dụng dựa trên ba tiêu chí sau cho đối tượng mục tiêu:

* Hiệu quả - Độ chính xác và đầy đủ mà người dùng được chỉ định có thể đạt được các mục tiêu được chỉ định trong một môi trường cụ thể.
* Hiệu quả - Các tài nguyên sử dụng liên quan đến tính chính xác và đầy đủ của các mục tiêu đạt được.
* Sự hài lòng - Sự thoải mái và chấp nhận của hệ thống làm việc đối với người dùng và những người khác bị ảnh hưởng bởi việc sử dụng nó.
* Phải kiểm tra khả năng sử dụng ngày từ giai đoạn đầu của thiết kế ứng dụng và không nên chỉ thực hiện khi ứng dụng hoàn thành

Kịch bản kiểm thử:

* Kiểm tra phòng trống nên có trên trang chủ.
* Quảng cáo được tài trợ không nên được hiển thị ở giữa nội dung
* Đảm bảo rằng các nút phải có kích thước yêu cầu và phù hợp với ngón tay.
* Đảm bảo rằng các nút được đặt trong cùng một phần của màn hình để tránh gây nhầm lẫn cho người dùng cuối.
* Đảm bảo rằng các biểu tượng là tự nhiên và phù hợp với ứng dụng.
* Đảm bảo rằng các nút có cùng chức năng cũng phải có cùng màu.
* Đảm bảo rằng việc xác thực cho các tiện ích phóng to và thu nhỏ phải được bật.

##### Kiểm thử bảo mật

* Lỗ hổng để hack, xác thực và chính sách thẩm định, bảo mật dữ liệu, quản lý data và tiêu chuẩn bảo mật khác cần được xác nhận đó là một phần của Security Testing.
* Một cách để kiểm tra các trường hợp liên quan đến an ninh là để lộ dữ  
  liệu của điện thoại di động của bạn thông qua một máy chủ proxy như  
  OWASP Zed Attack Proxy và tìm kiếm các lỗ hổng

Kịch bản kiểm thử:

* Chắc chăn rằng các ứng dụng không được thực thi với các thông tin của cùng một người dùng trên hai thiết bị di động khác nhau.
* Chắc chắn rằng một phiên làm việc tự động hết hạn, nếu nó vẫn không hoạt động trong thời gian hơn 15 phút.



Hình ảnh 2.3: Hình ảnh minh họa về kiểm thử bảo mật

##### Kiểm thử hiệu suất

* Thực hiện kiểm thử hiệu suất của Mobile bao gồm hiệu suất của ứng dụng client, hiệu suất server và hiệu suất mạng.
* Mục tiêu cơ bản của loại thử nghiệm này là đảm bảo ứng dụng hoạt động có thể chấp nhận được theo các yêu cầu hiệu suất nhất định như truy cập bởi một số lượng lớn người dùng hoặc loại bỏ một phần cơ sở hạ tầng quan trọng như máy chủ cơ sở dữ liệu.

Kịch bản kiểm thử:

* Xác minh rằng việc check về các chuyến bay chuyến bay có sẵn chỉ mất một khoảng thời gian hợp lý.
* Xác minh rằng trong khi đang check các chuyến bay có sẵn điện thoại di động hoạt động bình thường và không treo.
* Xác định mức tiêu thụ pin của ứng dụng
* Xác định hiệu suất ứng dụng khi thực hiện thay đổi mạng từ Wifi thành 2G/3G và ngược lại



Hình ảnh 2.4: Hình ảnh minh họa về kiểm thử hiệu suất

##### Kiểm thử độ chịu tải

- Stress Testing là trường hợp kiểm thử với mục đích phát hiện lỗi trong những trường hợp ngoại lệ, bị treo, và deadlocks mà có thể không được chú ý trong quá trình Functional Testing và interface testing.

Dưới đây là danh sách của một số các tiêu chí để Stress Testing:

* Tải ứng dụng của bạn với dữ liệu càng nhiều càng tốt để cố gắng đạt được điểm giới hạn của nó.
* Thực hiện các hành động tương tự và nhiều hơn thế nữa.
* Thực hiện các hoạt động lặp đi lặp lại ở tốc độ khác nhau - rất nhanh hoặc rất chậm.
* Để ứng dụng của bạn chạy trong một thời gian dài, cả hai tương tác với các thiết bị và chỉ để cho nó nghỉ sau khi thực hiện một số nhiệm vụ tự động sau một thời gian dài, ví dụ: một slideshow.
* Tap ngẫu nhiên vào màn hình và bàn phím thiết bị trong quá trình chạy ứng dụng.
* Cài đặt nhiều ứng dụng chạy trên thiết bị của bạn để bạn có thể chuyển đổi giữa các ứng dụng của bạn và các ứng dụng khác

Kịch bản kiểm thử

* Kiểm tra 1000 người dùng đang truy cập vào các ứng dụng điện thoại di động để tìm kiếm các chuyến bay nội địa.
* Kiểm tra 1000 người dùng đang truy cập vào các ứng dụng điện thoại di động để tìm kiếm các chuyến bay quốc tế

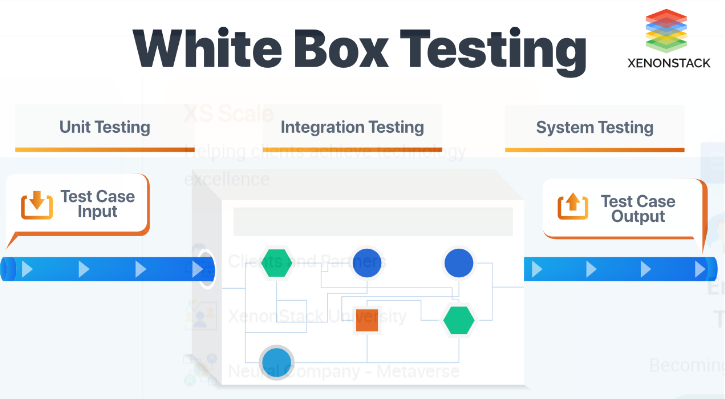


Hình ảnh 2.5: Hình ảnh minh họa về kiểm thử độ chịu tải

#### Kĩ thuật kiểm thử phần mềm

##### Kiểm thử hộp trắng (White box testing)

Kiểm thử hộp trắng dựa vào thuật toán, cấu trúc code bên trong của chương trình với mục đích đảm bảo rằng tất cả các câu lệnh sẽ được thực hiện ít nhất một lần.



Hình ảnh 2.6: Kiểm thử hộp trắng

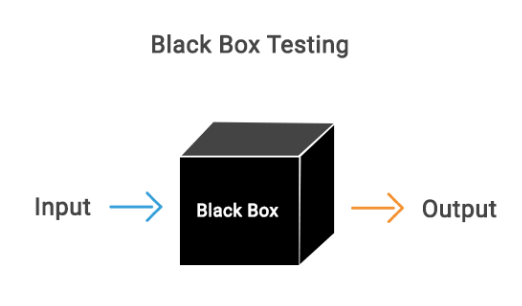
* Kiểm thử hộp trắng: là kỹ thuật kiểm thử dựa trên đặc tả bên trong của chương trình, dựa vào mã nguồn, cấu trúc chương trình.
* Kiểm thử hộp trắng thường phát hiện các lỗi lập trình, dễ thực hiện chi phí không cao.
* Với các module quan trọng, thực thi việc tính toán chính cảu hệ thống, phương pháp này là cần thiết.

Các kỹ thuật kiểm thử dùng trong kiểm thử hộp trắng:

* Bao phủ câu lệnh - Statement Coverage (SC).
* Bao phủ rẽ nhánh - Bao phủ quyết định Branch Coverage.
* Bao phủ đường dẫn - Path Throught Coverage

##### Kiểm thử hộp đen (Black box testing)

Kiểm thử hộp đen là phương pháp kiểm thử phần mềm mà việc kiểm tra các chức năng của một ứng dụng không cần quan tâm vào cấu trúc nội bộ hoặc hoạt động của nó. Mục đích chính của kiểm tra hộp đen chỉ là kiểm tra xem phần mềm hoạt động có đúng như dự kiến trong tài liệu têu cầu và liệu nó có đáp ứng được sự mong đợi của người dùng hay không.



Hình ảnh 2.7: Kiểm thử hộp đen

Đối tượng kiểm thử: Là thành phần phần mềm có thể là một hàm chức năng, một module chức năng, một phân hệ chức năng.

Phương pháp thử nghiệm: Dựa vào chức năng kiểm thử hộp đen (Black box test) có thể áp dụng hầu như đến mọi cấp độ của kiểm thử phần mềm:

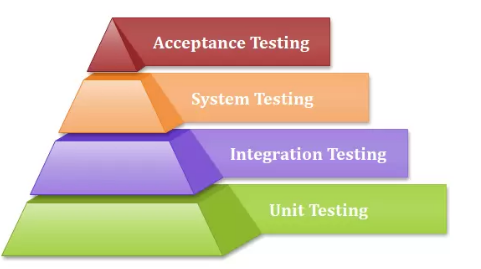
* Kiểm thử đơn vị (Unit test)
* Kiểm thử tích hợp (Intergration test)
* Kiểm thử hệ thống (System test)
* Kiểm thử chấp nhận (Acceptance test)

#### Các cấp độ kiểm thử phần mềm

Cấp độ kiểm thử phần mềm hay mức độ kiểm thử phầm mềm là một quá trình trong đó mọi thành phần của phần mềm hệ thống đều được kiểm tra. Thông qua các mức độ kiểm thử phần mềm sẽ giúp chúng ta đánh giá được chức năng của ứng dụng được những yêu cầu đã chỉ định hay chưa. Đồng thời thông qua mức độ kiểm thử phần mềm sẽ giúp chúng ta tìm và sửa lỗi nhằm đảm bảo sản phẩm tạo ra có chất lượng tốt nhất.

Tất cả các giai đoạn của quá trình phát triển phầm mềm đều trải qua bốn mức kiểm thử phần mềm là:

* Kiểm thử đơn vị (Unit Testing) - Dev
* Kiểm thử tích hợp (Integration Testing) - Dev
* Kiểm thử hệ thống (System Testing) – Tester
* Kiểm thử chấp nhận (Acceptance Testing) – Tester



Hình ảnh 2.8: Các cấp độ kiểm thử phần mềm

**Mức độ 1: Kiểm thử đơn vị - Unit Test**

Kiểm thử đơn vị - Unit Testing là giai đoạn đầu tiên trong kiểm thử phần mềm. Với chức năng hoạt động đơn giản, không gây nhiều khó khăn trong việc kiểm thử, ghi nhận và phân tích kết quả do đó nếu phát hiện lỗi thì việc tìm kiếm nguyên nhân và sửa lỗi cũng đơn giản và tốn ít chi phí hơn. Tuy nhiên, kiểm thử mức đơn vị lại tốn nhiều thời gian để thực hiện, chưa phát hiện được các lỗi xảy ra khi tích hợp.

**Mục tiêu:**

* Xác định mỗi đơn vị phần mềm có đang thực hiện theo đúng thiết kế ban đầu hay không.
* Thông qua thử nghiệm sẽ giúp khắc phục những phát sinh do việc thay đổi hay bảo trì code.
* Unit Testing giúp tiết kiệm chi phí, thời gian và thể diện khi phát hiện ra lỗi.

**Mức độ 2: Integration Testing – Kiểm thử tích hợp**

Mỗi dự án phần mềm được hoàn thành bởi rất nhiều module do nhiều người code khác nhau. Integration Testing là cấp độ kiểm thử phần mềm tích hợp của các đơn vị riêng lẻ được kết hợp và thử nghiệm thành một nhóm thông qua việc tập trung vào kiểm tra truyền dữ liệu giữa các module.

**Mục tiêu:**

* Phát hiện lỗi giao tiếp xảy ra giữa các Unit cũng như lỗi của bản thân từng Unit
* Tích hợp các Unit đơn lẻ thành các hệ thống nhỏ và cuối cùng là nguyên hệ thống hoàn chỉnh chuẩn bị cho kiểm tra ở mức hệ thống

**Một số phương pháp kiểm thử tích hợp:**

* Phương pháp kiểm thử Bigbang
* Phương pháp kiểm thử Topdown
* Phương pháp kiểm thử Bottom up
* Phương pháp kiểm thử Sandwich

**Mức độ 3: System Testing – Kiểm thử hệ thống**

System Testing là giai đoạn thứ 3 của kiểm thử phần mềm cho phép phần mềm hoàn chỉnh và tích hợp được kiểm tra. System Testing tập trung nhiều hơn vào các chức năng của toàn bộ hệ thống. Kiểm thử hệ thống bao gồm kiểm thử chức và kiểm thử phi chức năng

**Mục tiêu:**

* System Test kiểm tra thiết kế và toàn bộ hệ thống sau khi tích hợp có tuân thủ những yêu cầu đã được định sẵn trước đó.
* System Test kiểm thử cả các hành vi chức năng của phần mềm lẫn các yêu cầu về chất lượng như độ tin cậy, tính tiện lợi khi sử dụng, hiệu năng và bảo mật.

**Mức độ 4: Acceptance Testing – Kiểm thử chấp nhận**

Sau khi kiểm tra hệ thống đã sửa tất cả hoặc hầu hết các lỗi, hệ thống sẽ được gửi đến người dùng hoặc khách hàng để kiểm tra chấp nhận. Về cơ bản kiểm thử chấp nhận cũng khá giống kiểm thử hệ thống nhưng được thực hiện bởi khách hàng

Mục đích của Acceptance Testing đó là xác nhận lại sự tin tưởng vào hệ thống, các đặc tính thuộc về chức năng hoặc phi chức năng của hệ thống.

Có 2 loại kiểm thử chấp nhận đó là Alpha Testing và Beta Testing.

* Kiểm thử alpha: được thực hiện tại nơi phát triển phần mềm bởi những người trong tổ chức nhưng không tham gia phát triển phần mềm.
* Kiểm thử beta: được thực hiện tại bởi khách hàng/ người dùng cuối tại địa điểm của người dùng cuối.

### File:Unofficial JavaScript logo 2.svg - Wikimedia CommonsNgôn ngữ Javascipt

Hình ảnh .: JavaScript logo

JavaScript là một ngôn ngữ lập trình đa nền tảng (cross-platform), ngôn ngữ lập trình kịch bản, hướng đối tượng. JavaScript là một ngôn ngữ nhỏ và nhẹ (small and lightweight). JavaScript chứa các thư viện tiêu chuẩn cho các đối tượng (object), ví dụ như: Array, Date, Math, và các yếu tố cốt lõi của ngôn ngữ lập trình như: toán tử (operators), cấu trúc điều khiển (control structures), và câu lệnh. JavaScript có thể được mở rộng cho nhiều mục đích bằng việc bổ sung thêm các object; ví dụ:

* Client-side JavaScript - JavaScript phía máy khách, JavaScript được mở rộng bằng cách cung cấp các object để quản lý trình duyệt và Document Object Model (DOM) của nó. Ví dụ, phần mở rộng phía máy khách cho phép một ứng dụng tác động tới các yếu tố trên một trang HTML và phản hồi giống các tác động của người dùng như click chuột, nhập form, và chuyển trang.
* Server-side JavaScript - JavaScript phía máy chủ, JavaScript được mở rộng bằng cách cung cấp thêm các đối tượng cần thiết để để chạy JavaScript trên máy chủ. Ví dụ, phần mở rộng phía server này cho phép ứng dụng kết nối với cơ sở dữ liệu (database), cung cấp thông tin một cách liên tục từ một yêu cầu tới phần khác của ứng dụng, hoặc thực hiện thao tác với các tập tin trên máy chủ.

JavaScript được phát triển bởi Brendan Eich, một nhân viên của Netscape, vào tháng 9 năm 1995. Được biết đến lần đầu tiên là Mocha, rồi sau đó được đổi tên thành LiveScript và cuối cùng là JavaScript nổi tiếng như hiện giờ. Phiên bản đầu tiên của ngôn ngữ này bị giới hạn độc quyền bởi Netscape và chỉ có các tính năng hạn chế, nhưng nó tiếp tục phát triển theo thời gian.

Năm 1996, JavaScript được chính thức đặt tên là ECMAScript. ECMAScript 2 phát hành năm 1998 và ECMAScript 3 tiếp tục ra mắt vào năm 1999. Nó liên tục phát triển thành JavaScript ngày nay, giờ đã hoạt động trên khắp mọi trình duyệt và trên khắp các thiết bị từ di động đến máy tính bàn.

JavaScript liên tục phát triển kể từ đó, có lúc đạt đến 92% website đang sử dụng JavaScript vào năm 2016. Chỉ trong 20 năm, nó từ một ngôn ngữ lập trình riêng trở thành công cụ quan trọng nhất trên bộ công cụ của các chuyên viên lập trình web.

JavaScript cùng với HTML, CSS đã trở thành ngôn ngữ không thể thiếu trong lập trình web hiện nay:

* Với việc phát triển mạnh mẽ của công nghệ, JavaScript đã dần trở thành một ngôn ngữ đa nền tảng với nhiều ứng dụng khác nhau.
* Các Framework, Library cho lập trình front-end mạnh mẽ hỗ trợ xây dựng các trang Single-page application (SPA) như: ReactJS, Angular, VueJS.
* Hỗ trợ lập trình mobile bằng thư viện React Native.
* Hỗ trợ lập trình Window bằng Electron.
* Không chỉ phát triển ở khía cạnh lập trình phía client-side, JavaScript cũng hỗ trợ viết server-side rất mạnh mẽ với NodeJS.

Với hệ sinh thái mã nguồn mở khổng lồ cùng sự phát triển như hiện nay, JavaScript được dự đoán sẽ phát triển mạnh mẽ hơn nữa trong những năm tới.

### Extension

#### Khái niệm

Extension là một phần mở rộng của trình duyệt web, cung cấp các tính năng mới và tùy chỉnh cho trình duyệt. Extension cho phép người dùng mở rộng tính năng của trình duyệt web theo cách mà nhà phát triển không cung cấp.

Các extension có thể thêm tính năng mới vào trình duyệt như chặn quảng cáo, lưu trữ mật khẩu, phát hiện ngôn ngữ, kiểm tra chính tả và nhiều tính năng khác. Các extension có thể được phát triển bởi các nhà phát triển độc lập hoặc bởi các công ty và tổ chức lớn.

Các trình duyệt web phổ biến như Google Chrome, Mozilla Firefox, Safari và Microsoft Edge đều hỗ trợ extension. Người dùng có thể tìm kiếm và cài đặt các extension từ các cửa hàng ứng dụng của trình duyệt. Các extension cũng có thể được tải xuống và cài đặt trực tiếp từ trang web của nhà phát triển.

Tuy nhiên, các extension có thể làm giảm hiệu suất của trình duyệt và làm giảm tính bảo mật của hệ thống. Do đó, người dùng nên cẩn thận khi cài đặt các extension từ các nguồn không đáng tin cậy và chỉ cài đặt các extension từ các nhà phát triển và cửa hàng ứng dụng được đáng tin cậy.

#### Kiến trúc của extension

Kiến trúc của một extension thường gồm các thành phần cơ bản như sau:

* Manifest: Đây là một tệp JSON (JavaScript Object Notation) chứa thông tin cơ bản về extension, bao gồm tên, phiên bản, mô tả, quyền truy cập, icon, đường dẫn đến tệp popup, v.v. Manifest được sử dụng để định nghĩa cấu hình và tính năng của extension.
* Background Page: Là một trang web ẩn được tạo ra bởi extension, nó có thể thực hiện các tác vụ đa tiến trình như xử lý các yêu cầu mạng, quản lý dữ liệu và lưu trữ. Background Page cũng cung cấp API để tương tác với các thành phần khác của trình duyệt.
* Popup: Là một cửa sổ xuất hiện khi người dùng nhấp chuột vào biểu tượng của extension trên thanh công cụ của trình duyệt. Popup có thể chứa các nút điều khiển hoặc cung cấp thông tin cho người dùng.
* Content Script: Là một mã JavaScript được chèn vào các trang web để thực hiện các tác vụ như tìm kiếm và thay thế, thêm hoặc xóa các thành phần HTML hoặc CSS. Content Script có thể truy cập và tương tác với các thành phần của trang web bằng cách sử dụng các API được cung cấp bởi trình duyệt.
* Options Page: Là một trang web được sử dụng để cung cấp cài đặt và tùy chọn cho extension.
* Injected Script: Là một mã JavaScript được chèn vào các trang web bởi Content Script để tương tác với các thành phần của trang web.
* Native Host: Cho phép extension tương tác với các ứng dụng máy tính bên ngoài như các ứng dụng định dạng tệp.

A picture containing text, screenshot, diagram, design

Description automatically generated

Hình ảnh 2.10: Kiến trúc của Extension

Các thành phần của extension này hoạt động cùng nhau để cung cấp các tính năng và chức năng cho người dùng trên trình duyệt.

#### Ngôn ngữ lập trình xây dựng extension

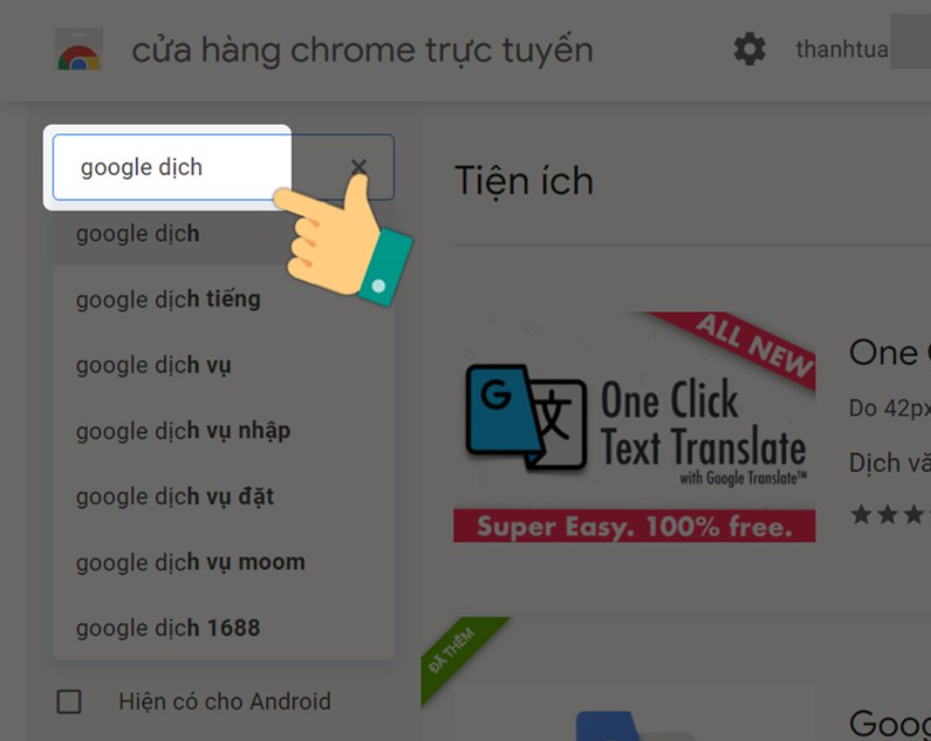
Có nhiều ngôn ngữ lập trình khác nhau có thể được sử dụng để xây dựng extension cho các ứng dụng khác nhau. Dưới đây là một số ngôn ngữ phổ biến được sử dụng để xây dựng extension:

1. C/C++: Đây là ngôn ngữ phổ biến nhất để xây dựng extension cho các ứng dụng như trình duyệt web, ứng dụng desktop, hệ điều hành và các phần mềm khác. C/C++ có thể được sử dụng để tạo các thư viện tĩnh hoặc động để mở rộng tính năng của ứng dụng.
2. Java: Java là một ngôn ngữ lập trình phổ biến để phát triển các ứng dụng web và desktop. Nó cũng có thể được sử dụng để phát triển các extension cho các ứng dụng Java, nhưng việc phát triển extension Java yêu cầu kiến thức chuyên sâu về Java.
3. Python: Python là một ngôn ngữ lập trình thông dịch phổ biến và được sử dụng rộng rãi trong phát triển web, khoa học dữ liệu và trí tuệ nhân tạo. Nó cũng có thể được sử dụng để phát triển các extension cho các ứng dụng desktop và web.
4. JavaScript: JavaScript là một ngôn ngữ lập trình phổ biến trong phát triển web. Nó có thể được sử dụng để phát triển các extension cho các trình duyệt web, nhưng việc phát triển extension JavaScript yêu cầu kiến thức chuyên sâu về HTML, CSS và JavaScript.
5. Swift: Swift là ngôn ngữ lập trình phổ biến để phát triển các ứng dụng iOS và macOS. Nó cũng có thể được sử dụng để phát triển các extension cho các ứng dụng này.

Tùy thuộc vào ứng dụng cụ thể mà một ngôn ngữ lập trình sẽ phù hợp hơn với việc phát triển extension. Trong đề tài nghiên cứu này nhóm em sử dụng HTML, CSS và JavaScript để xây dựng extension.

#### Cách tìm và cài đặt extension trên trình duyệt

**Bước 1:** Truy cập [Cửa hàng Chrome trực tuyến](https://chrome.google.com/webstore/category/extensions?hl=vi" \t "_blank). **Nhập các từ khóa liên quan đến tiện ích muốn thêm vào Google Chrome** tại thanh tìm kiếm.



Hình ảnh 2.11: Truy cập cửa hàng Chorme trực tuyến

**Bước 2:** Chọn Thêm vào Chrome tại tiện ích muốn thêm.

A picture containing text, screenshot, software, computer icon

Description automatically generated

Hình ảnh 2.12: Truy cập cửa hàng Chorme trực tuyến

**Bước 3:** Chọn Thêm tiện ích. Như vậy là đã thêm tiện ích thành công!

A picture containing text, screenshot, software, multimedia software

Description automatically generated

Hình ảnh 2.13: Thêm tiện ích

#### Cách gỡ, xóa bỏ tiện ích mở rộng trong Google Chrome

**Bước 1:** Để xóa tiện ích trên Chrome, hãy nhấn vào biểu tượng tiện ích trên góc phải màn hình trình duyệt Google Chrome.

A screenshot of a computer

Description automatically generated with medium confidence

Hình ảnh 2.14: Hiển thị Extension

**Bước 2:** Nhấn vào biểu tượng ba chấm sau tiện ích bạn muốn xóa.

A screenshot of a computer

Description automatically generated with medium confidence

Hình ảnh 2.15: Chọn tiện ích muốn xóa

**Bước 3:** Chọn Xóa khỏi Chrome

A screenshot of a computer

Description automatically generated with medium confidence

Hình ảnh 2.16: Xóa tiện ích

**Bước 4:** Xác nhận Xóa để gỡ bỏ tiện ích mở rộng này.

A picture containing text, screenshot, software, multimedia software

Description automatically generated

Hình ảnh 2.17: Xác nhận xóa tiện ích

## CHƯƠNG 3: THỰC NGHIỆM

### Kế hoạch thực nghiệm

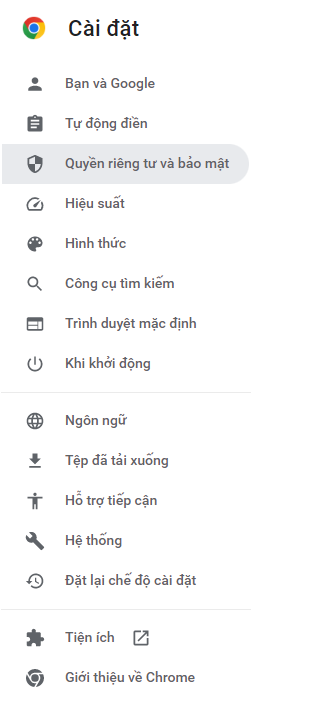
##### Kế hoạch

* Xây dựng extension chuyển đổi dữ liệu cơ bản

##### Mục đích

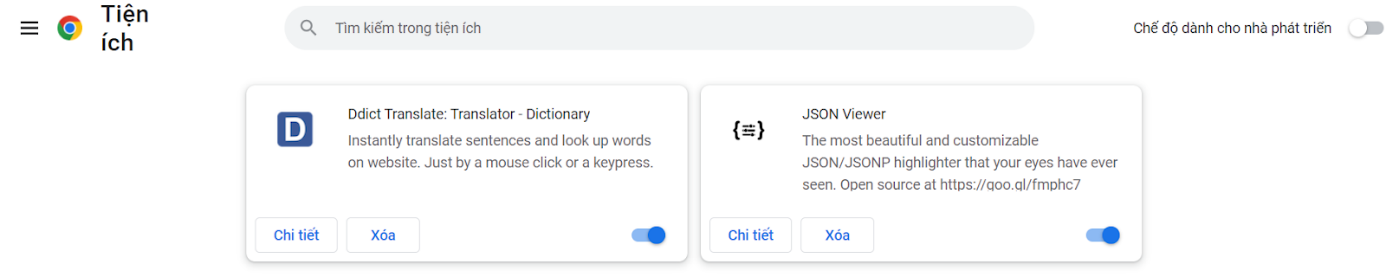
* Hiểu được các khái niệm cơ bản về Extension
* Cách crawl data từ Extension
* Cách tạo ra Extension chuyển đổi dữ liệu từ website này sang website khác

##### 3.1.3. Cách cài extension vào trình duyệt

****

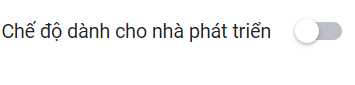
Hình ảnh 3.1: Mở cài đặt trên chorme

Ta vào cài đặt của chrome để vào “**Tiện ích”**



Hình ảnh 3.2: Chọn tiện ích

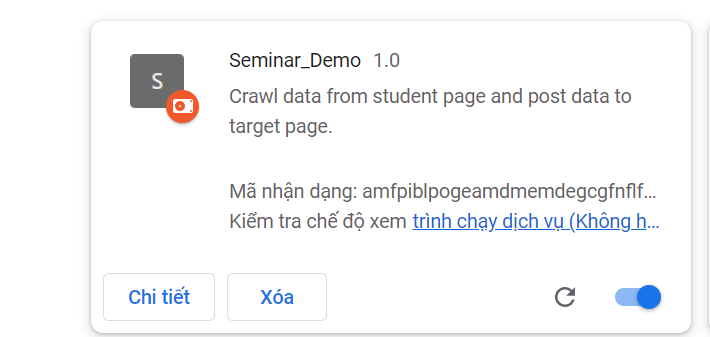
Sau đó bật chế độ nhà phát triển



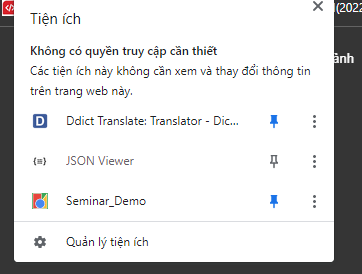
Hình ảnh 3.3: Bật chế độ nhà phát triển



Sau đó là tải tiện ích sau khi hoàn thiện và đã đóng gói. Mỗi khi có thay đổi bảo trì tiện ích ta sẽ cần reload lại Extension. Sau khi tải lên ta được như sau một ví dụ như sau và bật button cho phép sử dụng



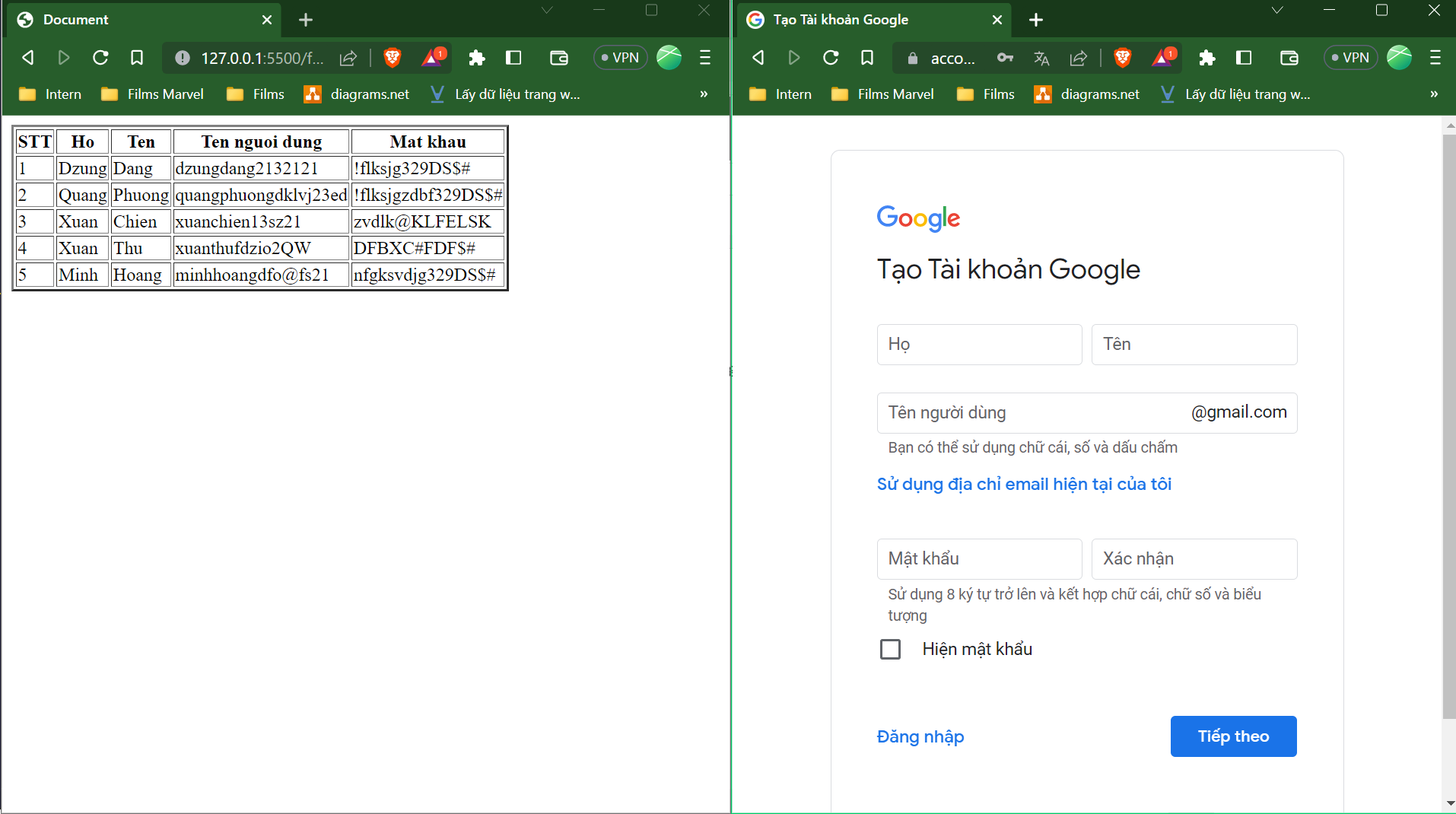
Hình ảnh 3.4: Cho phép xử dụng Extension



Hình ảnh 3.5: Ghim tiện ích mở rộng

### Thực nghiệm

Chúng ta có bảng bên trái chứa các dữ liệu như Họ, Tên, Tên người dùng, Mật khẩu, bảng bên phải là google form tạo tài khoản.



Hình ảnh 3.6: Tạo form tài khoản

Ở đây ta có trình duyệt đã cài sẵn extension chuyển đổi ở local.



Hình ảnh 3.7: Chọn extension sử dụng

Khi ta click vào extension thì dữ liệu trên được chuyển đổi qua google form đăng ký.

A screenshot of a computer

Description automatically generated

Hình ảnh 3.8: Chuyển đổi dữ liệu qua google form

Kết quả đạt được

### Giải thích hoạt động

Khi người dùng click chuột vào nút bấm trên extension, đoạn script sau sẽ được kích hoạt

A picture containing text, electronics, display, screenshot

Description automatically generated

Hình ảnh 3.9: Function sử lý sự kiện

A picture containing text, screenshot, font

Description automatically generated

Hình ảnh 3.10: Cấu hình URL

Trong đó, đoạn code trên thực hiện tìm kiếm các trang web có url khớp với giá trị http://127.0.0.1:5500/fakeData.html, trạng thái trang web đã load thành công.

A screenshot of a computer

Description automatically generated with medium confidence

Hình ảnh 3.11: Trích xuất tab

Thực hiện trích xuất ra 1 tab duy nhất.

A screenshot of a computer program

Description automatically generated with medium confidence

Hình ảnh 3.12: Thực thi script từ background

Thực thi script từ background lên content web của trang web được trích xuất phía trên (cụ thể là trang index có bảng dữ liệu).

Ở đây ta sẽ lấy thông tin cần thiết và lưu vào 1 object, sau đó ta bắn 1 message từ content web xuống background trình duyệt (chạy dưới nền), bên trong message có mã là ‘get-user-data’ và dữ liệu trong object được lấy từ content web.

Việc xử lý tiếp theo ta sẽ thực thi ở background, nếu message trả về từ response là ‘ok’ thì ta sẽ in ra màn hình dev-tool console như trên.

Tiếp theo ta sẽ đến với phần background scripts.

A picture containing text, electronics, display, screenshot

Description automatically generated

A screen shot of a computer

Description automatically generated with low confidence

Hình ảnh 3.13: Thực thi lắng nghe sự kiện

Ta sẽ thực hiện việc phản hồi message từ content web bên dưới background thông qua API chrome cung cấp như trên

A screen shot of a computer screen

Description automatically generated with low confidence

Hình ảnh 3.14: Kiểm tra message

Nếu message có giá trị là ‘get-user-data’, ta sẽ thực hiện xử lý theo logic phía dưới.

A screen shot of a computer screen

Description automatically generated with low confidence

Hình ảnh 3.15: Trả response cho content web

Trả về response cho content web với message là ‘ok’

Sau đó thực hiện query đến trang đăng kí tài khoản của google như sau

A screen shot of a computer code

Description automatically generated with medium confidence

Hình ảnh 3.16: Thực hiện quety đến trang đăng kí

Sau khi có được tab có url và trạng thái load thành công, ta sẽ thực thi nhúng code từ background lên trang web đó.

Việc thực thi sẽ thông qua API này.A screen shot of a computer screen

Description automatically generated with low confidence

Hình ảnh 3.17: Thực thi API

Mục tiêu tab mà script sẽ được thực thi là tabId lấy được từ câu lệnh query phía trên

A screen shot of a computer code

Description automatically generated with low confidence

Hình ảnh 3.18: Thực thi TabId

Việc thực thi nhúng scripts lên content web sẽ thực hiện bên trong method sau:

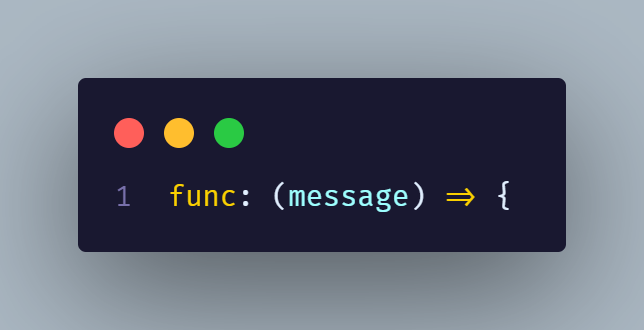
A screen shot of a computer code

Description automatically generated with low confidence

Hình ảnh 3.19: Thực thi query đến DOM

Cụ thể ta sẽ query đến các DOM element trên và sử dụng data được lấy từ object trong message để nhúng trực tiếp vào trang web đích.

Lưu ý, để có thể sử dụng data ở background bên trong ‘func’ method, ta cần truyền tham số vào như bên dưới



Hình ảnh 3.20: Truyền message

Và đồng thời đối số sẽ tương đương với tham số truyền vào như sau.



Kết thúc ta sẽ có kết quả như sau:

A screenshot of a computer

Description automatically generated

Hình ảnh 3.21: Kết quả thực thi

## CHƯƠNG 4: XÂY DỰNG EXTENSION CRAWL DATA TRONG ĐẠI HỌC ĐIỆN TỬ ĐẠI HỌC CÔNG NGHIỆP

### Mục đích

Mục đích để xây dựng extension crawl data trong Đại học điện tử Đại học Công Nghiệp nhằm hiểu được cách đưa được ứng dụng extension vào trong một nền tảng cụ thể để thực thi chức năng mà người phát triển đã cấu hình với mục đích cụ thể.

Chuyển đổi dữ liệu từ trên trang đại học điện tử về hệ thống của mình chuẩn đã chuẩn bị sẵn để hứng điểm của tài khoản đó.

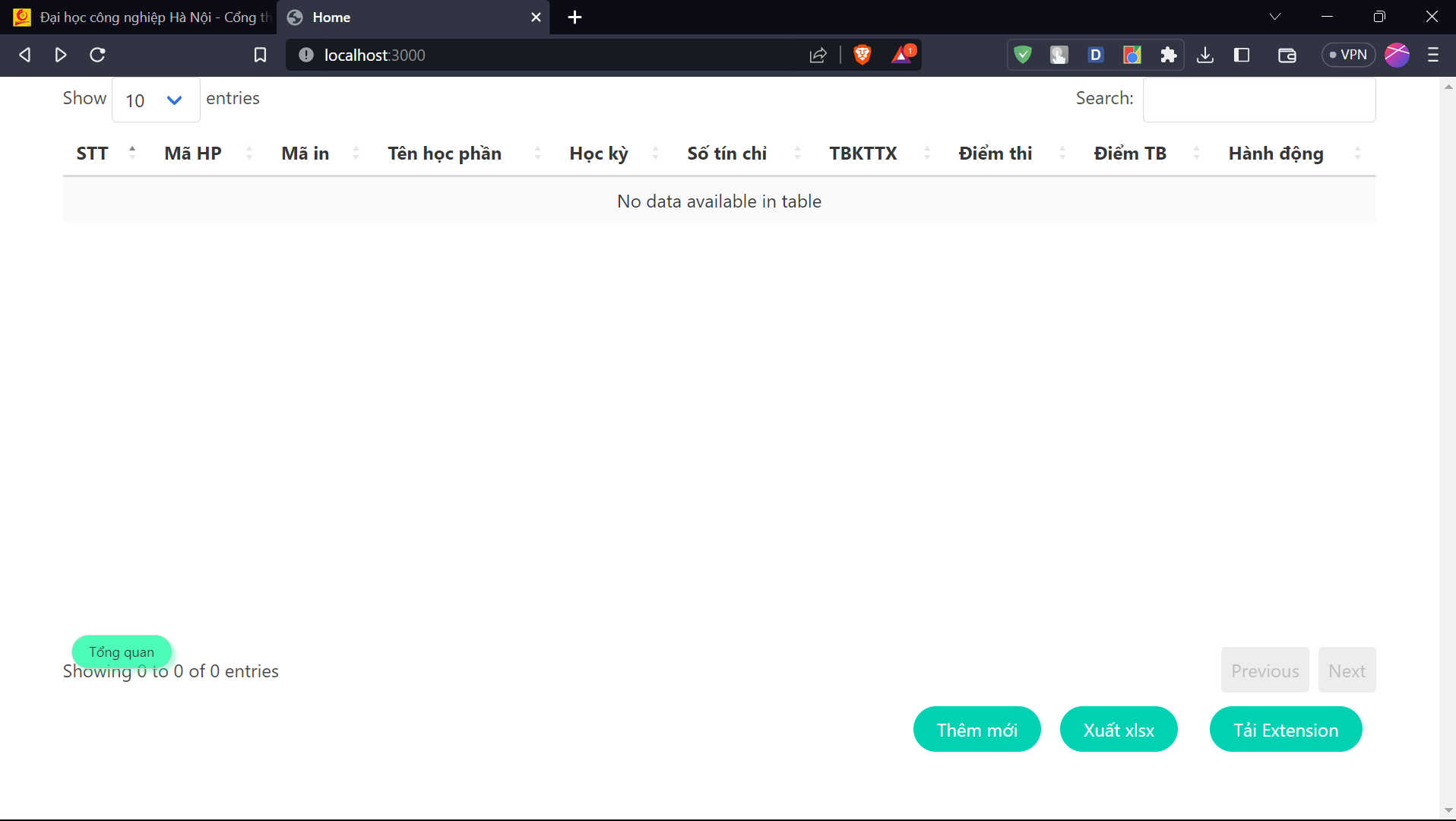
Phân tích điểm số từ bô dữ liệu đã được chuyển đổi

Ở đây để giúp cho sinh viên dễ dàng quản lý điểm của cá nhân. Với bộ dữ liệu cá nhân chính từ trang quản lý điểm của sinh viên trên đại học điện tử.

Từ đó tính toán số điểm mình cần đạt được để có kế hoạch học tập phù hợp với mục tiêu mình đề ra.

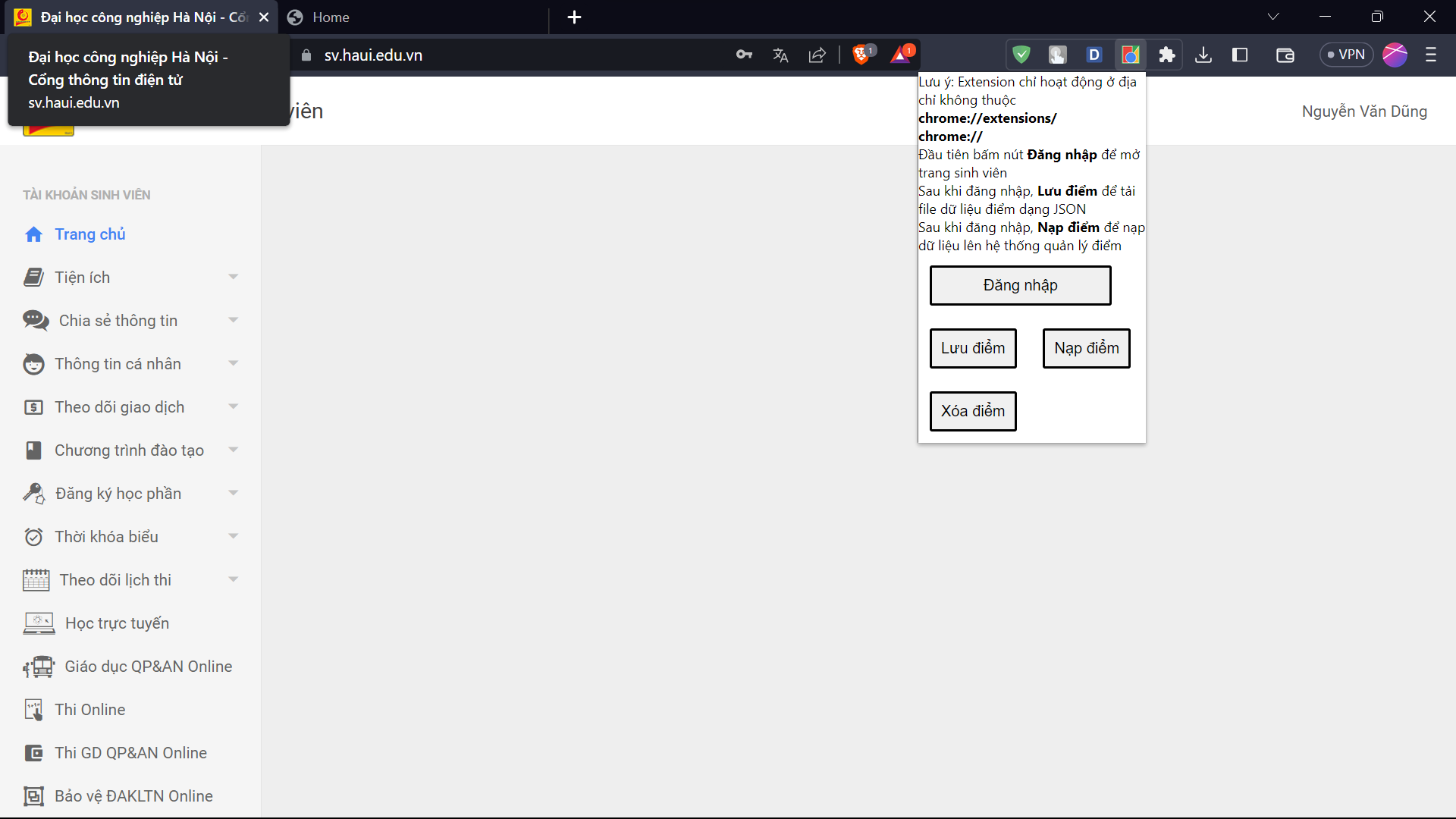
### Cài đặt ứng dụng

Ta cài đặt sẵn chương trình để mục đích để chứa dữ liệu sau khi crawl data. Để chuyển đổi thông tin bảng điểm từ hệ thống điểm trên trang sinh viên về máy local. Đầu tiên ta cần phải cài đặt Extension vào trình duyệt



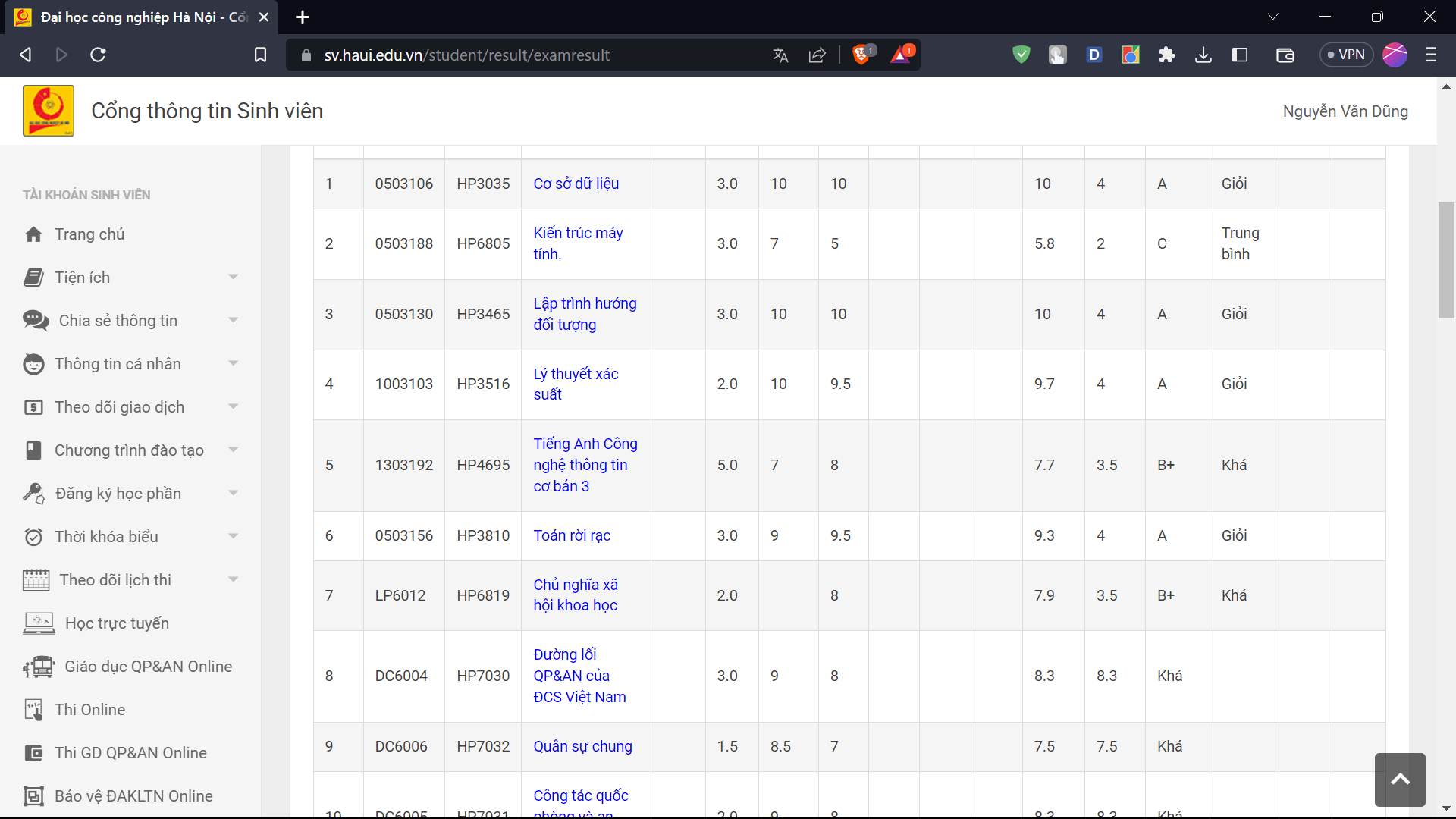
Hình ảnh 4.1: Hệ thống hứng dữ liệu

Sau khi cài đặt Extension vào trình duyệt ta có giao diện extension như sau:



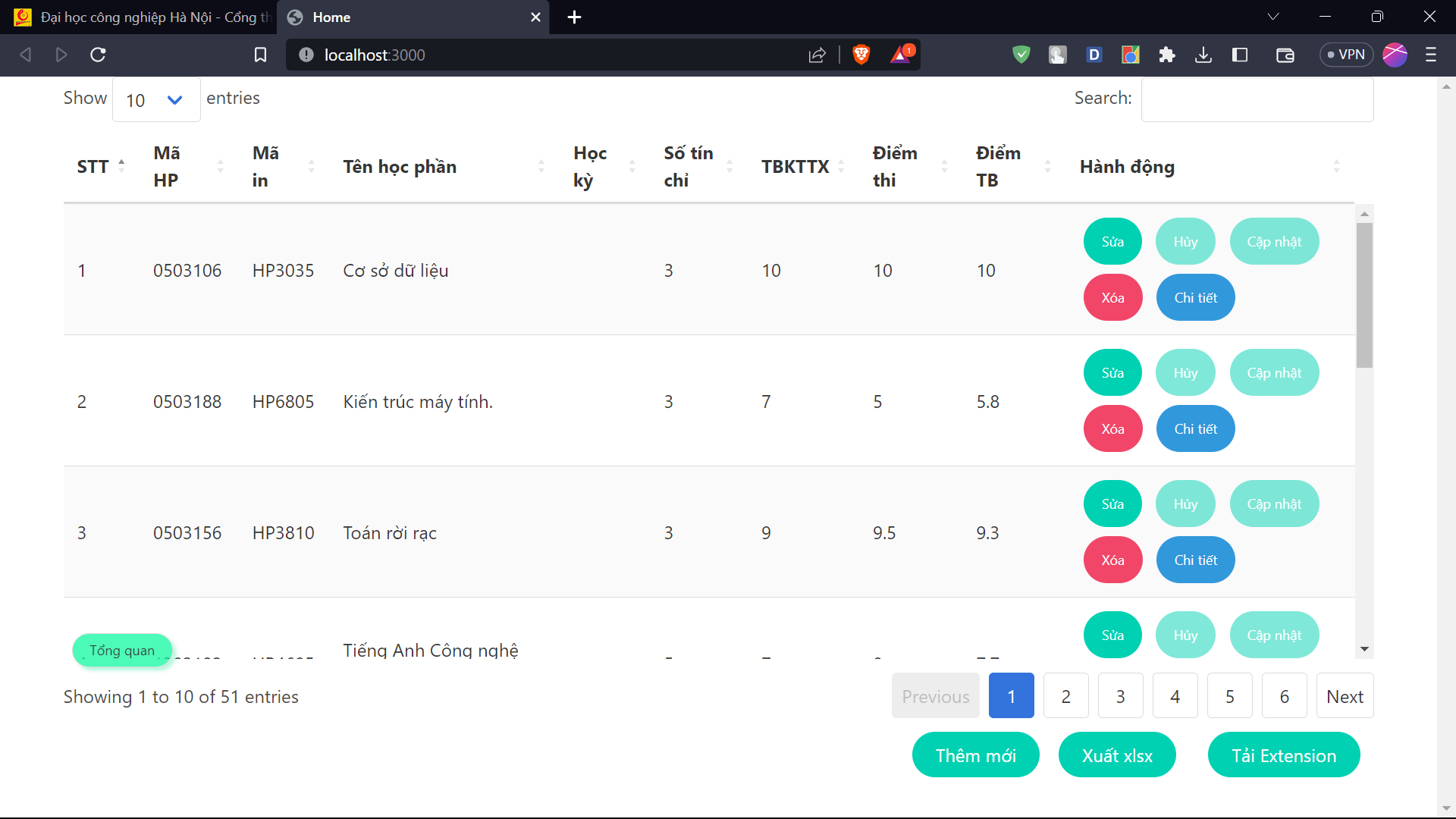
Hình ảnh 4.2: Cài đặt tiện ích craw**l** data

Sau khi Đăng nhập vào trang Sinh viên trình duyệt sẽ tự động điều hướng về trang quản lý điểm được dùng để chuyển đổi dữ liệu.



Hình ảnh 4.3: Hệ thống đại học điện tử

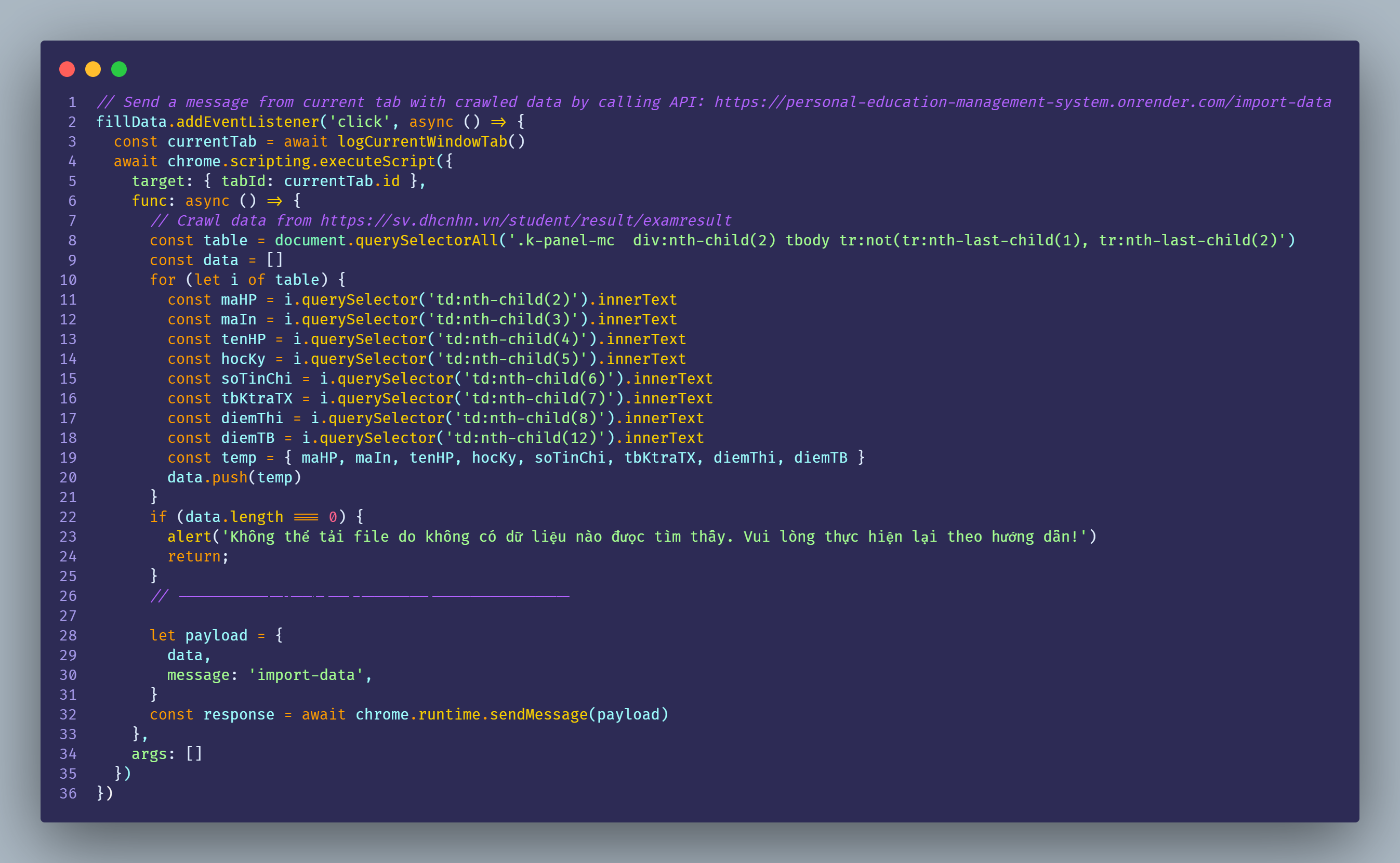
Sau khi ấn nút **Nạp điểm** thì trình duyệt local sẽ được nạp điểm thành công với kết quả như sau:



Hình ảnh 4.4: Kết quả craw**l** data

### Giải thích hoạt động

Quá trình chuyển đổi dữ liệu diễn ra như sau:



Hình ảnh 4.5: Thực thi sự kiện

Trên đây là code lắng nghe sự kiện khi người dùng Click chuột vào nút Nạp điểm ở trên giao diện Pop-up Extension. Extension sẽ tìm kiếm tab trình duyệt đang mở có đường dẫn là [https://sv.dhcnhn.vn/student/result/examresult](https://sv.dhcnhn.vn/student/result/examresult%20) để thực hiện nhúng Script (mã) Javascript.

Dựa trên cấu trúc trang web với các thẻ HTML từ cây DOM, ta thực hiện thu thập dữ liệu và lưu trữ dưới dạng mảng các môn học nằm trong biến `data`.

Để có thể thực hiện request HTTP/HTTPS từ phía extension chúng ta cần phải truyền dữ liệu xuống môi trường Background (nền) trình duyệt nơi mã JS của extension sẽ được thực thi.



Hình ảnh 4.6: Truyền dữ liệu

Việc truyền dữ liệu từ trang web (content) về trình duyệt nền (background) được thực hiện qua phương thức truyền/gửi tin nhắn 2 chiều.

Với message là loại tin nhắn, phía background sẽ nhận diện đâu là thông tin được gửi về từ phía trang web dựa trên yêu cầu.

Phía Background:

A picture containing text, screenshot, font, businesscard

Description automatically generated

Hình ảnh 4.7: Cấu hình chorm storage background

Trên đây là đoạn code lấy dữ liệu được lưu trữ lại ở chrome.storage background. Có thể coi đây là biến môi trường sử dụng cho từng người dùng cài đặt và sử dụng extension, được lưu trữ trong bộ nhớ background của trình duyệt.

A screen shot of a computer program

Description automatically generated with low confidence

Hình ảnh 4.8: Lắng nghe sự kiện

Đoạn mã trên cho thấy phía Background lắng nghe sự kiện được truyền về từ phía trang web, nếu có mã message là ‘import-data’ là mã message tự định nghĩa, background sẽ thực hiện gọi request HTTP đến server để thực hiện truyền dữ liệu thông qua request HTTP phương thức POST.

Việc thực hiện truyền dữ liệu sẽ gặp phải vấn đề về CORS (Cross-origin resource sharing), đây là một chính sách liên quan đến bảo mật được cài đặt vào toàn bộ các trình duyệt hiện nay. Chính sách này ngăn chặn việc truy cập tài nguyên của các domain khác một cách vô tội vạ. Vì thế extension cần phải có sự cho phép của server nếu muốn thực hiện gọi request HTTP/HTTPS đến server.

Bên cạnh đó, để có thể thực thi mã JS phía content web, chúng ta cần cấp quyền truy cập vào domain content web thông qua việc khai báo chúng bên trong manifest.json

A screenshot of a computer screen

Description automatically generated with medium confidence

Hình ảnh 4.9: Cấu hình manifest.json

# KẾT LUẬN

Kết luận:

Sau thời gian tìm hiểu và nghiên cứu về việc xây dựng một Extension trong Google chorm để tích hợp vào website chúng em nhận thấy những vấn đề đã đạt được sau:

* Hiểu các bước hình thành và xây dựng được Extension và tích hợp vào trong tiện ích mở rộng của google chorm dưới tư cách nhà phát triển.
* Ta
* Ứng dụng thành công trong việc tạo nên một Extension để đưa được được ứng dụng vào trang Đại học điện tử để chuyển đổi dữ liệu

Hạn chế:

Tuy nhiên, vẫn trong quá trình nghiên cứu cũng không thể tránh khỏi thiếu xót:

* Thiếu tài liệu nghiên cứu về vấn đề liên quan tới kiểm thử trong khi sử dụng Extension, chưa lên được kịch bản kiểm thử.
* Chưa hoàn thiện được kịch bản kiểm thử phần mềm trên nền tảng web

Hướng phát triển đề tài:

* Nghiên cứu thêm về các lý thuyết kiểm thử và xây dựng kịch bản kiểm thử trên nền tảng web
* Phát triển Extension có thể đưa vào được ứng dụng trong kịch bản kiểm thử trang web theo hướng giao diện, chức năng, …

# TÀI LIỆU THAM KHẢO

Tài liệu kiểm thử:

<https://www.guru99.com/automation-testing.html>

<https://www.guru99.com/difference-automated-vs-manual-testing.html>

<https://www.guru99.com/introduction-to-selenium.html>

<https://www.guru99.com/introduction-to-cucumber.html>

<https://viblo.asia/p/vi-sao-can-kiem-thu-tu-dong-Qpmlen4r5rd>

<https://digital.ai/catalyst-blog/what-is-appium-server-a-complete-end-to-end-guide>

Tài liệu Extension chorm:

<https://developer.chrome.com/docs/extensions/mv3/>

<https://topdev.vn/blog/cach-build-chrome-extension/> <https://anhtester.com/blog/chrome-extensions-huu-ich-danh-cho-tester-b302.html>

<https://developer.mozilla.org/en-US/docs/Web/API/Web_Workers_API>