TRƯỜNG ĐẠI HỌC CÔNG NGHỆ - ĐHQGHN KHOA CÔNG NGHỆ THÔNG TIN



NGUYỄN TIẾN ĐẠT

TÌM HIỀU VÀ XÂY DỰNG CÔNG CỤ KIỂM THỬ KHẢ NĂNG TIẾP CẬN CỦA ỨNG DỤNG WEB

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Khoa học máy tính

HÀ NỘI - 2022

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ - ĐHQGHN KHOA CÔNG NGHỆ THÔNG TIN



NGUYỄN TIẾN ĐẠT

TÌM HIỀU VÀ XÂY DỰNG CÔNG CỤ KIỂM THỬ KHẢ NĂNG TIẾP CẬN CỦA ỨNG DỤNG WEB

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Khoa học máy tính

Cán bộ hướng dẫn: TS. Võ Đình Hiếu

HÀ NỘI - 2022

Lời cảm ơn

Lời đầu tiên, tôi xin kính gửi lời cảm ơn sâu sắc tới thầy TS. Võ Đình Hiếu đã đưa ra đề tài, định hướng nghiên cứu, tận tình hướng dẫn và giúp đỡ tôi trong suốt quá trình thực hiện khóa luận này. Thầy đã tạo điều kiện tổ chức các buổi trao đổi hàng tuần để cung cấp kiến thức mới về kiểm thử khả năng tiếp cận của ứng dụng web cho sinh viên.

Tôi xin cảm ơn thầy/anh Nguyễn Bảo Ngọc đã luôn động viên, giúp đỡ và đưa ra những góp ý hữu ích cho tôi trong quá trình viết và hoàn thành khóa luận này.

Tôi xin chân thành cảm ơn các thầy, cô, cán bộ trong trường Đại học Công Nghệ - Đại học Quốc gia Hà Nội, đặc biệt là các thầy cô trong khoa Công Nghệ Thông Tin đã tạo mọi điều kiện thuận lợi nhất cho tôi học tập và nghiên cứu.

Cuối cùng, tôi cũng xin gửi lời cảm ơn sâu sắc đến tất cả các thầy cô, các anh chị, các bạn trong phòng thí nghiệm của bộ môn Công Nghệ Phần Mềm đã luôn hỗ trợ tôi rất nhiều về kiến thức chuyên môn trong quá trình thực hiện khóa luận này.

**Tóm tắt**

Tóm tắt: Trong vài năm gần đây,…

Từ khóa: …

**Lời cam đoan**

Tôi là Nguyễn Tiến Đạt, sinh viên lớp K63-CACLC3, ngành Công nghệ thông tin. Tôi xin cam đoan khóa luận này là những nghiên cứu mà tôi tự hiểu. Mọi thông tin tham khảo từ các tài liệu, công trình nghiên cứu liên quan đều được trích trong khóa luận này.Tôi xin chịu hoàn toàn mọi trách nhiệm về những lời cam đoan trên.

Hà Nội, ngày tháng 05 năm 2022

Sinh viên

Nguyễn Tiến Đạt

**Mở đầu**

Trong những năm gần, chuyển đổi kỹ thuật số và sự phát triển công nghệ mang lại lợi ích cho mọi người và cải thiện chất lượng cuộc sống của họ thông qua khả năng tiếp cận thông tin. Các ứng dụng web đóng vai trò quan trọng trong việc truyền tải thông tin tới mọi người một cách nhanh chóng. Do đó sự phát triển của ứng dụng web ngày càng nhận được sự quan tâm của cộng đồng. Hiện nay các ứng dụng web chưa thực sự đáp ứng được các tiêu chuẩn về khả năng tiếp cận, vẫn còn tồn tại nhiều lỗi tiếp cận ứng dụng web đối với người dùng. Điều này khiến một thị phần lớn thị trường người khuyết tật gặp khó khăn trong việc tiếp cận các ứng dụng web, theo WHO thị phần người khuyết tật chiếm 15% dân số thế giới.

Vì những tác hại của sự thiếu khả năng tiếp cận ứng dụng web, việc phát hiện và phân tích các lỗi trong khả năng tiếp cận ứng dụng web được xem là một yếu tố quan trọng nhằm giúp cho các ứng dụng web ngày càng phát triển và có thể được tiếp cận bởi nhiều người dùng hơn. Phát hiện ra các lỗi của khả năng tiếp cận ứng dụng web nhằm khắc phục sớm trong quá trình phát triển để đáp ứng được các tiêu chuẩn về khả năng tiếp cận của ứng dụng web trước khi được triển khai. Trong những năm gần đây, các công cụ phân tích và tìm ra lỗi về khả năng tiếp cận của ứng dụng web đã được phát triển. Tuy nhiên, các công cụ này chỉ phát hiện các lỗi tiếp cận ứng dụng web mà chưa đưa ra các gợi ý để sửa lỗi. Điều này khiến cho các nhà phát triển phải bỏ nhiều thời gian, công sức và gặp khó khăn trong việc khắc phục lại các lỗi tiếp cận của ứng dụng web. Việc xây dựng một công cụ kiểm thử khả năng tiếp cận ứng dụng web có chức năng gợi ý sửa lỗi sẽ đem lại hiệu quả tốt hơn.

Khóa luận này đề xuất bổ sung chức năng sửa lỗi cho công cụ kiểm thử khả năng tiếp cận ứng dụng web nhằm khắc phục lỗi nhanh chóng và hiểu quả hơn. Công cụ kiểm thử khả năng tiếp cận ứng dụng web mà khóa luận này xây dựng không chỉ cung cấp cho các lập trình viên thông tin chi tiết về các lỗi mà còn đưa ra cách sửa lỗi một cách trực quan, điều này giúp nhà phát triển, kể cả những người không có kiến thức về khả năng tiếp cận ứng dụng web có thể hiểu được và khắc phục các lỗi kịp thời.

Nội dung của khóa luận gồm 4 chương chính. Chương 1 trình bày các kiến thức tổng quan về kiểm thử khả năng tiếp cận ứng dụng web và các tiêu chuẩn trong kiểm thử khả năng tiếp cận ứng dụng web. Chương 2 trình bày về một số vấn đề phổ biến về khả năng tiếp cận ứng dụng web và các công cụ kiểm thử hiện tại. Từ các kiến thức nền tảng của chương 1 và kết quả nghiên cứu ở chương 2, chương 3 phân tích các điểm yếu của các công cụ hiện tại và trình bày phương pháp cải tiến công cụ kiểm thử khả năng tiếp cận ứng dụng web nhằm mục đích cải thiện các công cụ kiểm thử hiện có. Cuối cùng, chương 4 sẽ trình bày về cách xây dựng công cụ, những thử nghiệm và đưa ra những đánh giá về công cụ cải tiến mà khóa luận này xây dựng.

### **Chương 1 Kiến thức nền tảng**

* 1. Kiểm thử trong quy trình phát triển phần mềm
     1. Tổng quan
     2. Vai trò
  2. Kiểm thử khả năng tiếp cận

1.2.1 Khái niệm

1.2.2 Khuyết tật và khả năng tiếp cận

1.2.3 Các tiêu chuẩn trong kiểm thử khả năng tiếp cận

…..

**Chương 2 Vấn đề về khả năng tiếp cận của các trang web**

2.1 Một số vấn đề phổ biến

…..

2.2 Một số công cụ kiểm thử hiện tại

2.2.1 Wave

2.2.2 Taw

2.2.3 Achecker

**Chương 3 Xây dựng công cụ hỗ trợ kiểm thử khả năng tiếp cận trang web**

Chương này tiến hành phân tích các điểm yếu của các công cụ kiểm thử khả năng tiếp cận ứng dụng web hiện tại và trình bày phương pháp cải tiến công cụ, chi tiết về các module xử lý luồng nghiệp vụ để cải thiện các công cụ kiểm thử hiện tại.

**3.1 Vấn đề của các công cụ hiện tại**

Các ứng dụng web cần phải được kiểm thử trước khi triển khai và đi vào hoạt động, việc thực hiện đầy đủ các quy trình kiểm thử nhằm phát hiện ra lỗi có thể xảy ra trong hệ thống, từ đó nhà phát triển có thế khắc phục các lỗi đó một cách kịp thời. Trong đó kiểm thử khả năng tiếp cận ứng dụng web là một trong những khâu quan trọng trong quy trình kiểm thử. Tuy nhiên việc lựa chọn công cụ kiểm thử khả năng tiếp cận ứng dụng web phù hợp và hiệu quả cũng đem đến những sự khó khăn cho các công ty phát triển phần mềm. Tùy vào yêu cầu cũng như ngân sách của các công ty phát triển ứng dụng bên cạnh đó là một vài tiêu chí về các công cụ kiểm thử, từ đó các công ty có thể đưa ra sự lựa chọn công cụ phù hợp nhất.

1. Công cụ miễn phí và công cụ thương mại

Hiện nay trên thị trường có rất nhiều công cụ kiểm thử khả năng tiếp cận ứng dụng web, các công cụ này được chia làm hai loại là các công cụ có thể sử dụng miễn phí và loại thứ hai là các công cụ thương mại, nghĩa là các công ty hoặc cá nhân muốn sử dụng các công cụ đó phải mua bản quyền để có thể sử dụng chúng. Các công cụ miễn phí yêu cầu người dùng cần phải có nhiều kiến thức về tiếp cận ứng dụng web hơn, tuy nhiên việc đào tạo sử dụng các công cụ này lại tốn ít thời gian hơn do các chức năng thường khá đơn giản và dễ sử dụng. Chi phí phải trả để sử dụng công cụ gắn liền với quy mô kiểm thử ứng dụng mà nó đem lại. Các công cụ miễn phí chỉ kiểm tra được một trang web cho mỗi lần thực hiện trong khi đó các công cụ thương mại có thể kiểm tra, phân tích lỗi và thậm chí là cả sửa lỗi cả một ứng dụng web lớn. Việc thu thập thông tin kiểm thử của các ứng dụng web cũng rất quan trọng, nhà phát triển được yêu cầu phải có các báo cáo chi tiết về khả năng truy cập các ứng dụng web một cách cụ thể, và các báo cáo này chỉ được cung cấp bởi các công cụ kiểm thử bản thương mại.

1. Nền tảng công công cụ

Một số công cụ hoạt động trên nền tảng web, vì vậy người dùng không tải xuống hoặc cài đặt bất kỳ ứng dụng nào. Với giao diện web thân thiện người dùng có kiểm tra khả năng tiếp cận dễ dàng và nhanh chóng. Một số công cụ khác hoạt động trong trình duyệt. Chúng là các tiện ích mở rộng được tải xuống trên hầu hết các trình duyệt phổ biến và xuất hiện dưới dạng thanh công cụ hoặc menu bổ sung trong trình duyệt. Có những công cụ được nhúng trực tiếp vào quá trình tạo ứng dụng web, qua đó việc phát triển ứng dụng có thể được hợp nhất với kiểm tra khả năng tiếp cận. Ngoài ra còn có một số công cụ cần phải cài đặt trên phần cứng hoặc máy chủ. Những công cụ này hữu ích cho việc kiểm thử các ứng dụng web lớn và phức tạp.

1. Phạm vi

Phạm vi của một công cụ kiểm thử khả năng tiếp cận ứng dụng web được xác định bởi các tính năng của nó. Một số công cụ đánh giá, phân tích toàn bộ ứng dụng web và kiểm tra xem nó có tuân theo với các nguyên tắc về khả năng tiếp cận khác nhau không. Một số công cụ khác chỉ đánh giá khả năng truy cập của một trang duy nhất và đưa ra một báo cáo chi tiết về trang được kiểm tra. Ngoài ra còn có một số công cụ chỉ kiểm tra cụ thể một loại khả năng truy cập trong ứng dụng web. Ví dụ như kiểm tra tỷ lệ tương phản màu sắc, từ đó giải quyết các vấn đề liên quan đến loại khuyết tật cụ thể như mù màu, thị lực kém.

1. Công cụ chỉ đánh giá và công cụ đánh giá và sửa lỗi

Nhiều công cụ chỉ kiểm tra khả năng truy cập ứng dụng web mà không đưa ra có gợi ý để sửa lỗi, trong khi một số công cụ khác có thể đánh giá và sửa chữa các lỗi về khả năng truy cập từ đó giúp nhà phát triển rút ngắn thời gian để khắc phục các lỗi. Tính năng sửa chữa lỗi khả năng tiếp cận chỉ có trong các sản phẩm thương mại.

So sánh các công cụ kiểm thử khả năng tiếp cận

Khóa luận này chỉ tập chung nghiên cứu các công cụ kiểm thử khả năng tiếp cận miễn phí, độc lập và tuân theo tiêu chuẩn WCAG 2.0. Mốt số công cụ này được phát triển theo mã nguồn mở, từ đó giúp các nhà phát triển có thể chỉnh sửa, cập nhật và triển khai một cách linh hoạt. Cụ thể ba công cụ sẽ được khóa luận so sánh là Achecker, TAW và WAVE. Đây những những công cụ miễn phí, mã nguồn mở và được sử dụng rộng rãi hiện nay bởi các công ty phát triển phần mềm. Khóa luận tiến hành kiểm tra khả năng tiếp cận của các trang web khác nhau bằng cách sử dụng ba công cụ này. Các trang web được thử nghiệm có nội dung tiếng anh hoặc tiếng việt với các loại nội dung web khác nhau như là về tin tức, giải trí hay giáo dục. Cả ba công cụ này đều được sử dụng trên nền tảng web, người dùng chỉ cần truy cập vào mã định danh tài nguyên(URI) của mỗi công cụ để có thể sử dụng chúng. AChecker sau kiểm thử ứng dụng web tạo ra ba loại lỗi, cụ thể là lỗi xác định, lỗi có khả năng xảy ra và lỗi tiềm ẩn. Sau khi đánh giá, phân tích ứng dụng web WAVE đưa ra các loại lỗi, cảnh báo, tính năng, cấu trúc các phần tử của trang web và cả các lỗi tương phản. Các lỗi khả năng tiếp cận được phát hiện bởi TAW được phân loại thành lỗi xác định, cảnh báo và không được xem xét. Trong ba công cụ kiểm thử trên, công cụ duy nhất hiển thị các nguyên tắc đánh giá về khả năng tiếp cận được triển khai là AChecker trong khi TAW và WAVE ẩn các quy trình và nguyên tắc đánh giá khả năng truy cập web của chúng. Sau khi tiến hành kiểm thử khả năng tiếp cận của nhiều trang web khác nhau sử dụng Achecker, TAW và WAVE, kết quả kiểm thử được biểu diễn trong bảng 1 bên dưới.

|  |  |  |  |
| --- | --- | --- | --- |
| Website | Achecker | Taw | Wave |
| https://courses.uet.vnu.edu.vn/ | 178 | 90 | 16 |
| https://vnexpress.net/ | 425 | 546 | 68 |
| https://cafef.vn/ | 950 | 425 | 428 |
| https://www.w3schools.com/ | 557 | 387 | 63 |
| https://dantri.com.vn/ | 1574 | 40 | 261 |
| https://giaoduc.net.vn/ | 755 | 292 | 116 |
| https://hanoi.edu.vn/ | 642 | 280 | 106 |

Phân tích những kết quả này, ta có thể nhận xét rằng số lỗi được phát hiện ra bởi các công cụ kiểm thử là khác nhau đáng kể. Trong hầu hết các trang web, AChecker phát hiện ra số lượng lớn các vấn đề về khả năng truy cập hơn TAW và WAVE. TAW tạo ra các lỗi và hiển thị số lượng vi phạm theo nguyên tắc về khả năng tiếp cận. WAVE hiển thị snapshot của trang web và hiển thị lỗi vi phạm khả năng truy cập bằng các biểu tượng trên đó. Vì AChecker có hiệu suất tốt hơn trong việc phát hiện lỗi vi phạm khả năng tiếp cận ứng dụng web, khóa luận tập trung nghiên cứu về AChecker và các cách để cải thiện hiệu suất và tính tương tác của nó.

AChecker là một công cụ kiểm thử khả năng truy cập ứng dụng web được phát triển dưới dạng mã nguồn mở. Mục đích của công cụ này là phân tích khả năng truy cập của các ứng dụng web dựa trên nhiều loại của các nguyên tắc về khả năng tiếp cận của ứng dụng web quốc tế, chẳng hạn như Sec 508, BITV, STANCA, WCAG 1.0 và WCAG 2.0. Tuy nhiên khóa luận này chỉ tập trung vào nghiên cứu Achecker với tiêu chuẩn WCAG 2.0.

**3.2 Phân tích công cụ kiểm thử AChecker**

Các vấn đề của AChecker được phát hiện bằng cách tiến hành thử nghiệm kiểm tra các trang web khác nhau bằng công cụ AChecker và sử dụng cơ sở dữ liệu của nó để nghiên cứu các kĩ thuật đánh giá, phân tích lỗi ứng dụng web mà Achecker triển khai.

Các vấn đề của AChecker sau khi tiến hành thử nghiệm được phân loại thành ba nhóm. Nhóm thứ nhất là nhóm lỗi không rõ ràng, các lỗi được AChecker đưa ra một cách nhập nhằng, gây khó hiểu cho người dùng. Nhóm thứ hai là nhóm lỗi dư thừa, các lỗi Achecker phân tích được tồn tại nhiều lỗi trùng lặp với nhau. Nhóm thứ ba là nhóm lỗi thiếu sự gợi ý sửa lỗi, điều này khiến cho việc khắc phục lỗi trở lên khó khăn và tốn nhiều thời gian cho các nhà phát triển ứng dụng. Hơn nữa với những người không có kiến thức về khả năng tiếp cận ứng dụng web thì việc sửa lỗi gần như không thể thực hiện được. Sau khi tiến hành kiểm thử các trang web với Achecker, kết quả cho thấy các loại lỗi liên kết với phần tử của nó như bảng 2. Phần lớn các vấn đề được phát hiện liên quan đến vấn đề sửa lỗi.

|  |  |  |  |
| --- | --- | --- | --- |
| Phần tử web | Lỗi dư thừa | Lỗi không rõ ràng | Thiếu sửa lỗi |
| Image | ✓ | ✓ | ✓ |
| Link |  | ✓ |  |
| Frame |  |  | ✓ |
| Header |  | ✓ |  |
| Title |  |  | ✓ |
| Input | ✓ |  | ✓ |
| Table | ✓ |  | ✓ |
| Area code |  |  | ✓ |
| Html | ✓ |  | ✓ |
| Form | ✓ | ✓ |  |
| Anchor |  |  | ✓ |

Bảng 2.

Với thực trạng thực tế đã nêu ở trên, bài toán mà chúng ta cần giải quyết có 2 vấn đề:

1. Khắc phục lỗi không rõ ràng và lỗi trùng lặp xảy ra với công cụ Achecker.
2. Xây dựng thêm chức năng gợi ý sữa lỗi với giao diện trực quan. Qua đó mọi đối tượng người dùng có thể dễ dàng hiểu cách sửa lỗi cho từng trường hợp lỗi xảy ra mà không cần quá nhiều kiến thức chuyên môn, các nhà phát triển thì dễ dàng, đỡ tốn thời gian hơn trong quá trình khắc phục lỗi tiếp cận của ứng dụng web.

**3.3 Xây dựng công cụ cải tiến AChecker**

Để giải quyết vấn đề đã đặt ra, khóa luận xây dựng công cụ kiểm thử khả năng truy cập ứng dụng web MyChecker để cải thiện AChecker. Các lỗi được phát hiện bởi AChecker được phân loại theo các phần tử web có vấn đề về khả năng truy cập. Ví dụ các lỗi được phát hiện ra trên các thẻ như: image, link, frame, table, header, input, area, strong, label, keyboard, embedded, form. Khóa luận liệt kê một vài các vấn đề xảy ra trong các phần tử web này và đề xuất giải pháp được triển khai trong MyChecker.

1. Lỗi dư thừa

Vấn đề về các lỗi được công cụ kiểm thử Achecker tạo ra sau khi kiểm thử các trang web là sự trùng lặp nhau, một vài lỗi xuất hiện lặp đi lặp lại hoặc những lỗi chứa nội dung vi phạm khả năng tiếp cận với ý nghĩa gần như nhau, điều này khiến cho người dùng cảm thấy rối và khó chịu trong khi đọc và phân tích lỗi. Cách tiếp cận của khóa luận để khắc phục vấn đề này là hợp nhất các lỗi trùng lặp thành một lỗi rõ ràng, loại bỏ đi những lỗi dư thừa, không cần thiết. Ví dụ với phần tử input Achecker đưa ra hai lỗi với nội dung tương tự nhau là phần tử input thiếu nhãn liên kết. Công cụ Mychecker gộp chúng lại thành một lỗi rõ ràng, dễ hiểu với người dùng. Bảng 3 dưới đây hiện thị một số lỗi trùng lặp và được khắc phục bởi công cụ MyChecker.

|  |  |  |
| --- | --- | --- |
| Mã lỗi | AChecker | MyChecker |
| 48 | Document language not identified | Language of the page is not specified |
| 49 | Document has invalid language code |
| 57 | Input element, type of "text", missing an associated label | The input element doesn't have associated label |
| 213 | Input element, type of "text", label has no text in label |
| 241 | Tabular information may be missing table markup | The information are not organized well |
| 248 | Visual lists may not be properly marked |
| 246 | All required form fields may not be indicated as required | Required fields are not specified |
| 267 | Form submission error messages may not identify empty required fields |
| 118 | Input element, type of "password", missing an associated label | The input element doesn't have associated label |
| 207 | Input element, type of "password", has no text in label |
| 120 | Input element, type of "file", missing an associated label | The input element doesn't have associated label |
| 216 | Input element, type of "file", has no text in label |
| 121 | Input element, type of "radio", missing an associated label | The input element doesn't have associated label |
| 204 | Input element, type of "radio", has no text in label |
| 119 | Input element, type of "checkbox", missing an associated label | The input element doesn't have associated label |
| 206 | Input element, type of "checkbox", has no text in label |
| 95 | Textarea element missing an associated label | The textarea element doesn't have associated label |
| 212 | Textarea has no text in label |

Bảng 3.

1. Lỗi không rõ ràng

Các lỗi không rõ ràng xảy ra khi mô tả lỗi không phù hợp hoặc khó hiểu. Cách tiếp cận MyChecker để cải thiện AChecker là thay thế những mô tả nhập nhàng, gây khó hiểu này bằng một mô tả đơn giản và rõ ràng mà bất kỳ người sử dụng nào cũng có thể hiểu được. Ví dụ với các lỗi với phần tử header, công cụ Achecker đưa ra thông báo về các lỗi này với nội dung là “phần tử header được sử dụng cho định dạng”, điều này khiến người đọc cảm thấy rất mơ hồ và khó hiểu. MyChecker sửa chúng thành “các phần tử header được sử dụng cho chia các phần trong nội dung trang web không phải được sử dụng để định dạng” giúp người dùng dễ hiểu hơn. Ngoài ra với nội dung lỗi “giá trị thuộc tính alt quá dài xảy ra trong phần tử img” thì người sử dụng cũng rất mơ hồ, không biết độ dài bao nhiêu là phù hợp. MyChecker thông báo lỗi với số lượng kí tự cụ thể cho giá trị thuộc tính “alt” điều này giúp cho người dùng biết cách chỉnh sửa lại nội dung sao cho phù hợp với độ dài yêu cầu. Bảng 4 dưới đây hiện thị một số lỗi không rõ ràng và được khắc phục bởi công cụ MyChecker.

|  |  |  |
| --- | --- | --- |
| Mã lỗi | AChecker | MyChecker |
| 1 | Image Alt text may be too long | Limit of text in alt attribute is 100 characters |
| 42 | H1 may be used for formatting | Headers should be used for separating section not using for formatting |
| 43 | H2 may be used for formatting | Headers should be used for separating section not using for formatting |
| 44 | H3 may be used for formatting | Headers should be used for separating section not using for formatting |
| 45 | H4 may be used for formatting | Headers should be used for separating section not using for formatting |
| 46 | H5 may be used for formatting | Headers should be used for separating section not using for formatting |
| 47 | H6 may be used for formatting | Headers should be used for separating section not using for formatting |
| 268 | Form submission error messages may not provide assistance | Provide help for correcting  the submitted errors |
| 38 | Header nesting - header following h2 is incorrect | H4 follow h2, headers should be used in order |

Bảng 4.

1. Sửa lỗi

Các lỗi được Achecker phân tích từ các ứng dụng web đưa ra là những đoạn văn bản cùng với những kiến thức liên quan về khả năng tiếp cận. Do đó để có thể khắc phục được lỗi, người dùng phải kinh nghiệm, kiến thức chuyên môn về kiểm thử khả năng truy cập ứng dụng web. Để giải quyết vấn đề này, công cụ MyChecker đề xuất chức năng sửa lỗi với giao diện và tính tương tác trực quan giúp các nhà phát triển thì dễ dàng, đỡ tốn thời gian hơn trong quá trình khắc phục lỗi tiếp cận của ứng dụng web. Lỗi thiếu thuộc tính “alt” xảy ra với phần tử img, MyChecker sẽ tự động thêm thuộc tính “alt” vào phần tử img đồng thời giá trị của thuộc tính này được nhập tùy chỉnh bởi người dùng. Khi ngôn ngữ không được chỉ định trong ứng dụng web, AChecker tạo ra hai lỗi trùng lặp với nội dung tương tự nhau: “ngôn ngữ nội dung trang web chưa được xác định”. Công cụ MyChecker khắc phục lỗi này bằng cách tự động thêm thuộc tính “lang” vào phần tử html và mã ngôn ngữ được nhập bởi người dùng đồng thời MyChecker cũng giải quyết vấn đề về lỗi dư thừa do AChecker tạo ra. Dựa trên tiêu chuẩn WCAG 2.0, phần tiêu đề là cần thiết cho mỗi trang web. AChecker tạo ra lỗi vi phạm khả năng tiếp cận trong hai trường hợp liên quan đến tiêu đề trang web. Trong trường hợp không có thẻ tiêu đề, MyChecker sẽ trực tiếp sửa lỗi bằng cách thêm thẻ tiêu đề với nội dung tiêu đề được nhập bởi người dùng vào trang web. Trong trường hợp nội dung tiêu đề để trống hoặc chứa nội dung không hợp lệ, công cụ MyChecker sẽ sửa lỗi bằng cách cho phép người dùng chỉnh sửa lại nội dung của thẻ tiêu đề. WCAG 2.0 đề cập rằng mỗi phần tử đầu vào phải có một nhãn liên quan để mô tả mục đích của đầu vào. AChecker tạo ra hai lỗi trong trường hợp thiếu nhãn được liên kết. MyChecker đưa ra giải pháp bằng cách thêm phần tử nhãn và cho phép người dùng nhập nội dung nhãn liên kết đề mô tả chi tiết về mục đích của đầu vào một cách rõ ràng hơn. Bảng 5 dưới đây hiện thị một số lỗi được khắc phục bởi công cụ MyChecker.

|  |  |  |
| --- | --- | --- |
| Mã lỗi | Mô tả lỗi | MyChecker |
| 1 | Phần tử img thiếu thuộc tính “alt” | Tự động thêm thuộc tính “alt” và giá trị được nhập bởi người dùng |
| 7 | Phần tử img với thuộc tính “alt” không hợp lệ | Cho phép người dùng nhập lại giá trị thuộc tính “alt” |
| 58 | Phần tử img được dùng trong input thiếu thuộc tính “alt” | Tự động thêm thuộc tính “alt” và giá trị được nhập bởi người dùng |
| 165 | Phần tử embed thiếu thuộc tính “alt” | Tự động thêm thuộc tính “alt” và giá trị được nhập bởi người dùng |
| 166 | Phần tử embed với thuộc tính “alt” không hợp lệ | Cho phép người dùng nhập lại giá trị thuộc tính “alt” |
| 101 | Phần tử frame với thuộc tính “alt” không hợp lệ | Cho phép người dùng nhập lại giá trị thuộc tính “alt” |
| 213 | Phần tử input không có nhãn liên kết | Tự động thêm nhãn cho phần tử input và giá trị được nhập bởi người dùng |
| 91 | Phần tử select không có nhãn liên kết | Tự động thêm nhãn cho phần tử select và giá trị được nhập bởi người dùng |
| 208 | Phần tử select với nhãn không hợp lệ | Cho phép người dùng nhập lại giá trị nhãn của phần tử liên kết |
| 212 | Phần tử area không có nhãn liên kết | Tự động thêm nhãn cho phần tử area và giá trị được nhập bởi người dùng |
| 207 | Phần tử input không có nhãn liên kết | Tự động thêm nhãn cho phần tử area và giá trị được nhập bởi người dùng |
| 31 | Phần tử iframe không có thuộc tính “title” | Tự động thêm thuộc tính “title” và giá trị được nhập bởi người dùng |
| 201 | Phần tử iframe chứa thuộc tính “title” không hợp lệ | Cho phép người dùng nhập lại giá trị thuộc tính “title” |
| 190 | Phần tử anchor thiếu thuộc tính “title” | Tự động thêm thuộc tính “title” và giá trị được nhập bởi người dùng |
| 50 | Trang web thiếu tiêu đề | Tự động thêm phần tử title trong phần head và giá trị được nhập bởi người dùng |
| 51 | Trang web chứa tiêu đề không hợp lệ | Cho phép người dùng nhập lại giá trị phần tử title trong phần head |
| 48 | Ngôn ngữ nội dung trang web chưa được xác định | Tự động thêm thuộc tính “lang” vào thẻ html và giá trị được nhập bởi người dùng |
| 3 | Thuộc tính “alt” quá dài | Cho phép người dùng nhập lại giá trị thuộc tính “alt” và giới hạn độ dài là 100 kí tự |

Bảng 5.

Sau khi phân tích các vấn đề của công cụ kiểm thử khả năng tiếp cận ứng dụng web AChecker và đưa ra giải pháp cho MyChecker, phần thiết kế hệ thống MyChecker được trình bày sau đây.

**Thiết kế hệ thống MyChecker**

Phần này trình bày chi tiết về các thành phần của hệ thống đánh giá khả năng tiếp cận được đề xuất trong khóa luận MyChecker. Nó mô tả và giải thích các thành phần cần thiết trong hệ thống MyChecker. Ngoài ra các yêu cầu chức năng, phi chức năng của hệ thống và kiến trúc của nó cũng được miêu tả chi tiết và đầy đủ. Các yêu cầu chức năng được định nghĩa là các dịch vụ mà hệ thống phải thực hiện. Chúng được phân loại theo chức năng cụ thể của MyChecker. Các yêu cầu phi chức năng là các ràng buộc về an toàn, bảo mật, hiệu năng, mức độ khả dụng v.v mà hệ thống phải đáp ứng

1. Yêu cầu chức năng

Các chức năng chính của MyChecker là phân loại và sửa chữa các lỗi vi phạm khả năng tiếp cận ứng dụng web. Vì mục đích này, một cơ sở dữ liệu mới được xây dựng chứa các loại lỗi với giải pháp sửa chúng. Các lỗi vi phạm khả năng tiếp cận đã được nghiên cứu và phân loại thành mô-đun lỗi không rõ ràng, lỗi dư thừa và sửa chữa lỗi để nâng cao kết quả khả năng tiếp cận ứng dụng web. Công cụ MyChecker được xây dựng với các dịch vụ, chức năng có thể tương tác trên giao diện web tạo điều kiện thuận lợi cho việc kiểm tra khả năng tiếp cận. Ngoài ra khi các lỗi vi phạm khả năng tiếp cận được giải quyết, kết quả thu được có thể được xuất ra tệp hoặc in giúp người dùng có thể quản lý và lưu dữ liệu lỗi vi phạm một cách tốt hơn.

1. Yêu cầu phi chức năng

Các yêu cầu phi chức năng của MyChecker liên quan đến hiệu suất, hoạt động, độ tin cậy, khả năng bảo trì và khả năng tương tác, mức độ khả dụng. MyChecker được triển khai trên máy chủ node và cơ sở dữ liệu MySQL để truy xuất thông tin thuận tiện hơn. Ngoài ra, hệ thống phải cho phép nhiều người dùng có thể truy cập hệ thống đồng thời và người dùng không cần cài đặt bất kỳ phần mềm đặc biệt nào để sử dụng hệ thống. MyChecker hỗ trợ nhiều trình duyệt web khác nhau. Các yêu cầu phi chức năng khác được xem xét trong quá trình thiết kế và phát triển MyChecker được liệt kê bên dưới.

* Yêu cầu về hiệu năng: Thời gian phản hồi trung bình của hệ thống phải tương ứng với mức độ phức tạp của các đối tượng được kiểm tra.
* Yêu cầu về hoạt động: Ứng dụng có thể tạo ra các thông báo lỗi tự động trong trường hợp hệ thống trục trặc hay gặp sự cố bất ngờ.
* Yêu cầu về độ tin cậy: Các mô-đun trong hệ thống phải cung cấp các thông báo chẩn đoán trong trường hợp hoạt động của các chức năng thành công, không thành công hoặc xảy ra lỗi.
* Yêu cầu về khả năng bảo trì: Tất cả các mô-đun phần mềm được phát triển trong dự án sẽ được phát hành theo giấy phép mã nguồn mở.
* Yêu cầu về khả năng sử dụng: Khả năng sử dụng của hệ thống được tính toán theo hiệu quả và sự hài lòng về mặt định lượng và chất lượng.

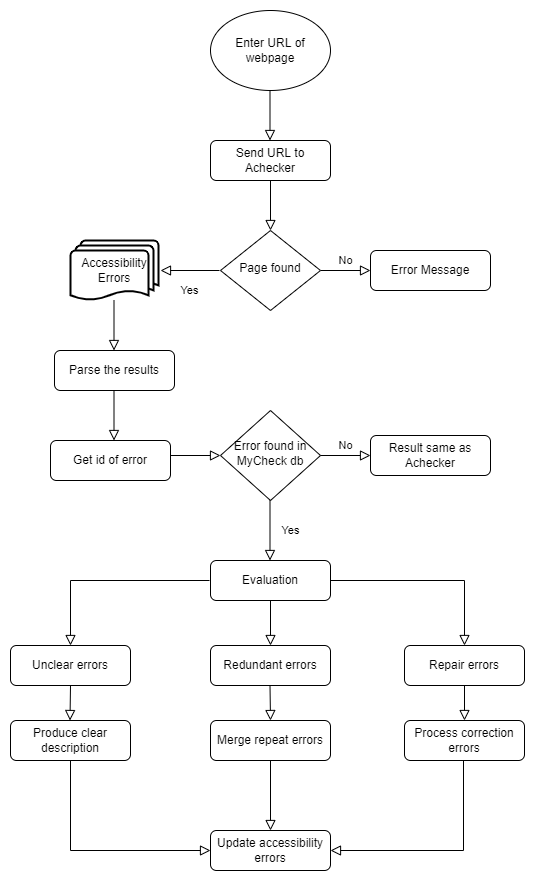
1. Kiến trúc hệ thống

Công cụ MyChecker là được xây dựng với năm mô đun chính bao gồm: mô đun phân tách kết quả từ AChecker, mô đun xử lý lỗi dư thừa, mô đun xử lý lỗi không rõ ràng, mô đun sửa lỗi và mô-đun kết quả. Mỗi một mô đun trong ba mô đun đầu tiên có một vai trò cụ thể trong quá trình đánh giá, phân tích khả năng tiếp cận của ứng dụng web. Trong khi hai mô đun sửa chữa và mô đun kết quả liên quan đến mục tiêu sửa chữa lỗi vi phạm khả năng tiếp cận. (H.1)

Trong mô đun phân tách kết quả, MyChecker giao tiếp với AChecker và nhận kết quả phân tích lỗi vi phạm khả năng tiếp cận ở định dạng HTML. MyChecker sử dụng thuật toán phân tích cú pháp từ các tệp kết quả lỗi HTML và lấy nội dung chi tiết của các lỗi từ các thẻ HTML rồi lưu chúng vào một mảng kết quả. Kết quả chứa ba thuộc tính là mã lỗi, mô tả lỗi và chi tiết lỗi. Mã lỗi xác định lỗi vi phạm khả năng truy cập bằng một số nhận dạng duy nhất. Mô tả lỗi là mô tả chi tiết về lỗi vi phạm theo WCAG 2.0. Chi tiết lỗi là mã html của trang web có vấn đề về khả năng truy cập. Mô đun lỗi không rõ ràng xử lý các lỗi vi phạm khả năng tiếp cận là các lỗi có mô tả mơ hồ, không rõ ràng và thay thế mô tả của chúng thành các từ đơn giản, dễ hiểu và rõ ràng. Mô đun lỗi dư thừa kết hợp các lỗi có sự trùng lặp để chỉ tạo ra một lỗi duy nhất giúp tránh sự nhầm lẫn, rõ ràng trong quá trình đọc và phân tích lỗi. Mô đun sửa lỗi của MyChecker là một cách tiếp cận mới giúp sửa lỗi vi phạm khả năng tiếp cận ứng dung web một cách hiệu quả hơn. Trong khi AChecker chỉ mô tả lỗi cũng như đề xuất cách khắc phục dựa trên văn bản, MyChecker sửa lỗi theo cách tương tác và cung cấp cho người dùng chức năng sửa lỗi một cách trực quan, dễ hiểu. Hơn nữa, quy trình sửa chữa được bán tự động để giúp các nhà phát triển dù không có nhiều kinh nghiệm về khả năng tiếp cận dễ dàng, đỡ tốn thời gian hơn trong quá trình khắc phục lỗi tiếp cận của ứng dụng web. Các đề xuất sửa chữa lỗi phải được viết bằng ngôn ngữ đơn giản để người sử dụng có thể hiểu được, cho dù họ có kiến thức về khả năng tiếp cận ứng dụng web hay không.

Mô đun kết quả lưu trữ các lỗi vi phạm khả năng tiếp cận gặp phải, mỗi lỗi được mô tả bằng id định danh của nó, mô tả lỗi và chi tiết về lỗi. Những kết quả này được hiện thị trên một giao diện tương tác của MyChecker. Hơn nữa, kết quả có thể được xuất thành một báo cáo ở định dạng excel hỗ trợ người dùng quản lý và lưu trữ dữ liệu về kết quả phân tích lỗi tốt hơn.

Hình 1 dưới đây là một sơ đồ cho thấy các quy trình thực thi của năm mô đun trong MyChecker và sự tương tác giữa của nó với AChecker.



Hình 1. Flow chart của MyChecker

**Chương 4 Triển khai và đánh giá công cụ**

**4.1 Tiến hành xây dựng công cụ**

**4.1.1 Backend**

**4.1.2 Fontend**

**4.2 Thử nghiệm và đánh giá công cụ**