

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 INTRODUCTION

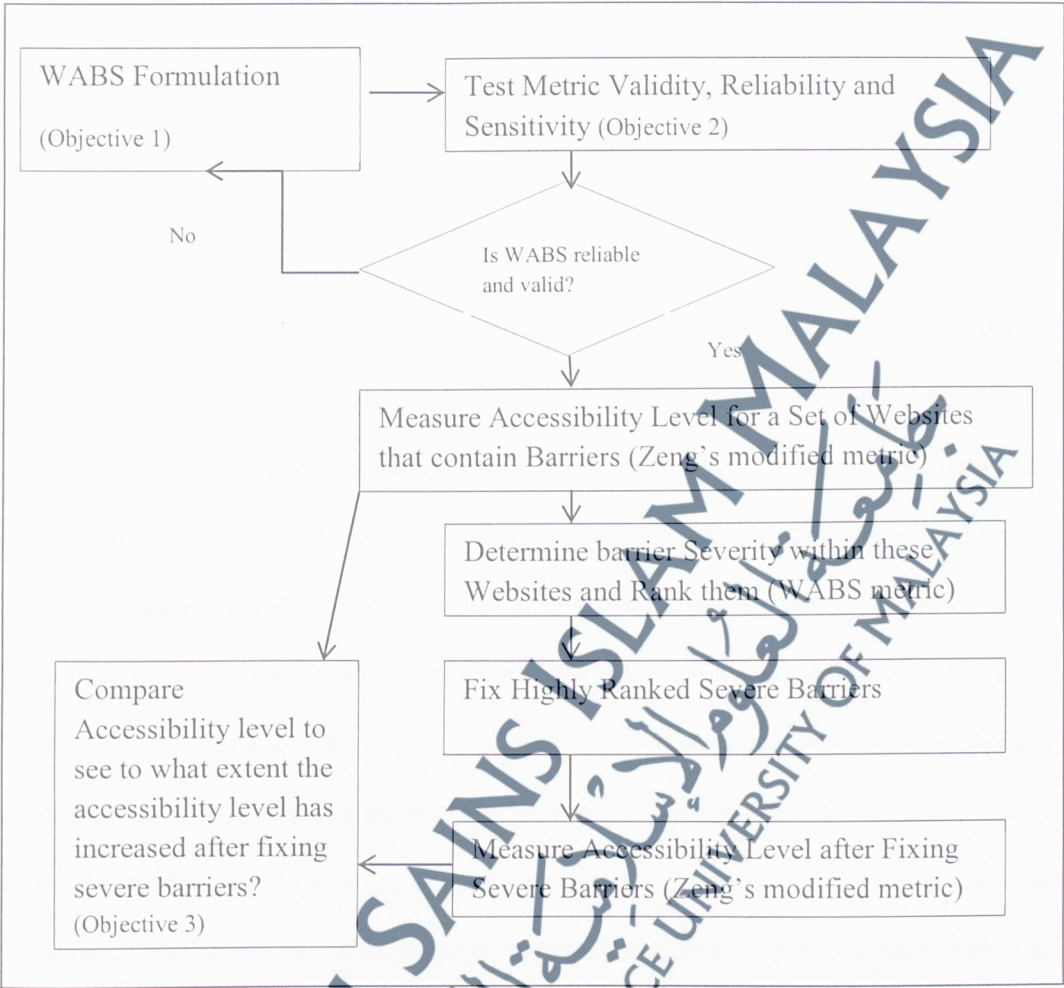
The research methodology and design process discussed in this chapter is based on the theoretical and conceptual framework explained in Chapter 2. This chapter discusses the process of validating the metric's quality and the process of measuring the degree of accessibility level improvement obtained by the metric. Section 3.2 describes the research design strategy which is divided into two main subsections (section 3.2.1 and 3.2.2). Section 3.2.1 shows the method for validating the metric quality. It describes the research hypotheses which are formulated to check the metric quality and how the hypotheses will be validated through experiments. Section 3.2.1.1 demonstrates the research approach for validating the metric quality. Meanwhile, section 3.2.2 shows the method for measuring the improvement degree of accessibility level when using the proposed metric and section 3.2.2.1 shows the approach for achieving that. Section 3.3 explicates the data collection and describes the websites dataset that will be used within the experiments. Section 3.4 describes the procedure for selecting the automated tools for the experiments. Finally, Section 3.5 concludes the chapter.

3.2 RESEARCH DESIGN

The first objective of this work is to propose a quantitative metric that ranks accessibility barriers based on their severities. However, the metric formulation to meet this objective

is discussed in a separate chapter which is Chapter 4. After the formulation of the metric which is called WABS, its quality has to be checked in terms of validity, sensitivity and reliability. This will enable meeting the second objective which is: evaluate the metric reliability and validity through empirical studies. If the metric proves that it meets the requirements of quality, the accessibility level for a set of Malaysian websites will be measured using the metric proposed by (Zeng, 2004) after modification. Following, WABS metric will be applied to this set in order to determine the barriers severity in that set and ranked them based on their severity. Then, the highly ranked severe barriers which are determined within the Malaysian websites will be removed and the accessibility level of websites will be measured again using Zen's modified metric to compare the accessibility level before and after fixing the highly severe barriers. This will enable fulfilling the third objective which aims to measure the degree of accessibility level improvement for a set of websites after fixing the top severe barriers detected by the metric. Figure 3.1 depicts the research process along with research objectives.

FIGURE 3.1: The Research Process



3.2.1 Validating Metric Quality

Several quality factors can be defined for web accessibility metrics. These factors can be used to assess how applicable a metric is in a certain context. Metric's validity, reliability, sensitivity, adequacy and complexity appear to be the most important factors (Vigo & Brajnik, 2014). This work focuses on the effectiveness of the metric. Effectiveness is defined as the extent to which the method can be used to yield the desired results with appropriate levels of accuracy and completeness (Brajnik (a), 2008). Effectiveness is

further refined into validity, sensitivity and reliability. Although all the qualities are important, this section emphasizes that validity, sensitivity and reliability of metrics should be given priority. It does not matter how complex or adequate a metric is, if one cannot ensure its reliability and especially validity.

Validation of software metric confirms or rejects the correctness of a given implementation of that particular software metric regarding its specification (Rüdiger & Löwe, 2006; Drouin et al., 2013; Srinivasan & Devi, 2014). That is, the calculated metric's values correspond to values which are expected by the metric specification. It provides information about mathematical and statistical operations that can be performed with the measure, which is very essential in this area. There are two types of validations for metrics: theoretical and empirical validations (Srinivasan & Devi, 2014). The theoretical validation confirms that the measurement does not violate any necessary properties of metric measurements. The empirical validation confirms that measured values of attributes are consistent with values predicted by models involving the attribute (Rüdiger, 2009). There are three types of empirical validations: surveys, case studies and experiments.

The theoretical validation of the WABS metric should test whether the metric adheres to its requirements and assumptions. For empirical validation, Sheppard & Darrel (1993) stressed the point that the measurement method should be capable of discrimination and will not assign the same value to every object. Another assumption that requires validation is: the more the barrier appears in a dataset the highly likely the barrier will be

more severe. Hence, the empirical validity should test the correlation analysis between barrier frequency and barrier weight.

Another measurement of metric quality is the sensitivity which is a measure of how changes in metric output are reflected in actual changes to any given website (Vigo & Brajnik, 2011). Experiments could be set up to conduct sensitivity analysis to figure out which kind of barriers had the largest effect and under which circumstances. Given that valid and reliable metrics are used, this could reveal which accessibility barriers would have a more or less strong impact on conformance or use. Different experiments should be conducted in different contexts (time span, tools and samples) to demonstrate the sensitivity of the proposed metric. However, the metric should not be very sensitive to small changes, since extremely sensitive metrics are not robust enough and can over-react to small changes in web contents.

The last measurement of interest for quality is metric reliability which is related to the reproducibility and consistency of scores (W3C (b), 2012), i.e. the extent to which they are the same when evaluations of the same webpages are carried out in different contexts and for different goals (Brajnik (a), 2008). The inherent inconsistency of unreliable metrics limits the ability of evaluators to predict metric behavior. In other words, if a metric is instable, it is very difficult to effectively use it as a tool of either analysis or comprehension. Different experiments should be conducted to examine metric behavior under different contexts such as evaluating with different automated tools, time span and samples. However, reliability of metrics very often depends on the reliability of the underlying testing tools, and hence different tools may produce different results on the

same pages (Fernandes et al., 2011). Reliability can also be affected by the time of evaluation. Some websites become highly dynamic over time, which could result in different contents and produce different types of barriers. This may have effect on barrier weights, but the overall ranking of barriers may stay the same (Anderson et al., 2000). Accordingly, it is necessary to measure whether barriers ranking significantly change when the metric is carried out in different context. This can help in defining the context where the metric is most reliable. That is to say, cautious actions should be taken regarding when to expect reliable behavior across tools, time and different samples.

3.2.1.1 Research approach for validating metric quality

This section shows the approach for testing the metric quality which is related to the second research question which is: “Does the metric meet the requirements of a valid and reliable metric?” In order to test the reliability, sensitivity and validity of the metric, it needs to be applied under different contexts. Thus, this question is refined into four sub-questions which are:

- i. To what extent does each barrier contribute to accessibility problems in a given evaluation set? What are the severity and the weight of each barrier when evaluating a specific dataset? Do the barriers which violate higher priority, score higher weight than the others? Does the barrier ranking depend only on the priority level?
- ii. Do barriers' weights significantly change when utilized by different automated tools? Are the metric's results, tool independent? If there is a significant change, to what extent does it affect the overall ranking?

- iii. Is there a significant change in barriers' weights when evaluating the same dataset over time? If yes, does this change affect the overall ranking?
- iv. Does barriers' weight significantly change when data are collected from different samples? Can the results of the metric be generalized or are they specific to each experiment?

Five hypotheses are formulated to help answer the previous sub-questions as stated in Table 3.1. Accordingly, four experiments are conducted to validate the hypotheses. As demonstrated by Table 3.1, the first experiment validates the first and second hypotheses by checking if the metric values have discriminating power and if the barrier frequency has a positive correlation with its weight. The first hypothesis [H1] aims to check the metric validity. The second hypothesis [H2] target both validity and sensitivity since it validates the metric measurements and check the ability of the metric to discriminate between accessibility barriers' severity that belong to the same priority level. The second, third and fourth experiments validate hypothesis [H3], [H4] and [H5], respectively, these hypotheses check the metric reliability as well as sensitivity under different context such different automated tools, time span and samples. Within each experiment, a dataset of websites will be evaluated automatically using automated tools in order to check the type of barriers that is found as well as their frequency. Following, WABS metric will be applied to the results of each experiment to calculate each barrier's weight then rank them based on their severity. Finally, the results will be quantitatively analyzed using descriptive and inferential statistics in order to validate the hypotheses.

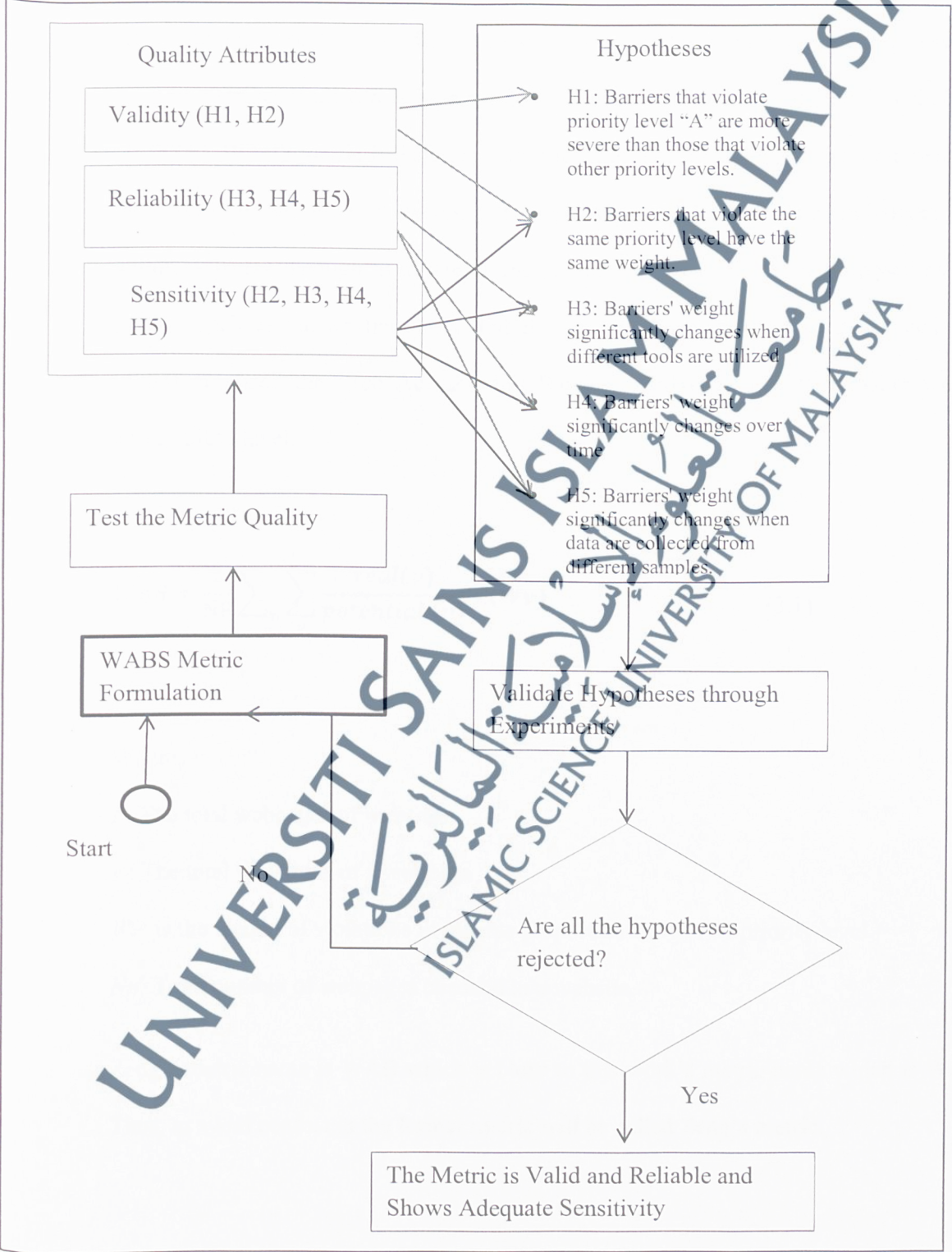
TABLE 3.1: Research Sub-Questions, Hypotheses and Experiments

Research Sub-Question	Hypothesis Number	Hypothesis	Quality Measurement	Experiment
To what extent does each barrier contribute to accessibility problems in a given evaluation set? What are the severity and the weight of each barrier when evaluating a specific dataset? Do the barriers which violate higher priority, score higher weight than the others? Does the barrier ranking depend only on the priority level?	H1	Barriers that violate priority level “A” are more severe than those that violate other priority levels.	Validity	Experiment 1: Measure the severity of each barrier
To what extent does each barrier contribute to accessibility problems in a given evaluation set? What are the severity and the weight of each barrier when evaluating a specific dataset? Do the barriers which violate higher priority, score higher weight than the others? Does the barrier ranking depend only on the priority level?	H2	Barriers that violate the same priority level have the same weight.	Validity & sensitivity	Experiment 1: Measure the severity of each barrier
Do barriers' weights significantly change when utilized by different automated tools? Are the metric's results, tool independent? If there is a significant change, to what extent does it affect the overall ranking?	H3	Barriers' weight significantly changes when different tools are utilized	Reliability & Sensitivity	Experiment 2: Check barriers' weight when evaluating by different tools
Is there a significant change in barriers' weights when evaluating the same dataset over time? If yes, does this change affect the overall ranking?	H4	Barriers' weight significantly changes over time	Reliability & Sensitivity	Experiment 3: Check barriers' weight when evaluating over time by the same tool
Does barriers' weight significantly change when data are collected from different samples? Can the results of the metric be generalized or are they specific to each experiment?	H5	Barriers' weight significantly changes when data are collected from different samples.	Reliability & Sensitivity	Experiment 4: Check barriers' weight when data are collected from different samples by the same tool
<i>Note:</i> WCAG 2.0 has Success Criteria (SCs) organized around three priority levels: A (must be satisfied), AA (should be satisfied), and AAA (may be satisfied). More information on priorities can be found in Table 2.2, while Appendix A has the full list of guidelines and SCs				

Figure 3.2 shows the process for testing metric quality. It is anticipated that testing these five hypotheses would assist in answering the second research question. If the hypotheses are rejected then the metric meets the quality’s requirements for building a good metric and can be used as a measurement for scientific research.



FIGURE 3.2: The Process of Testing WABS Quality



3.2.2 Measuring the Improvement in Accessibility Level

As mentioned in section 2.4, different metrics were proposed to measure the accessibility level of websites (Sullivan & Matson, 2000; Zeng, 2004; Arrue et al., 2005; Velleman et al., 2006). WABS metric is different from the other metrics since it focuses on measuring the severity of barriers in websites. It does not rank websites but common barriers found within websites. Though, it utilizes some parameters from the metric proposed by Zeng (2004), such as barrier frequency and the priority level that has been violated. Zeng (2004) proposed the Web Accessibility Barrier (WAB) formula to measure websites accessibility level.

$$WAB = \frac{1}{NP} \sum_p \sum_v \frac{real(v)}{potential(v)} * (Wv) \quad (3.1)$$

Where,

P : The total webpages of websites

v : The total violations of a webpage

Wv : is the weight of violations in inverse proportion to WCAG priority level.

Np : Total number of webpages checked in a website.

Zeng's metric name is WAB which is close to this work's metric name which is WABS.

Thus, to avoid confusion the former metric will be called Zeng's metric.

Although, the returned ratings of Zeng's metric are not restricted to a limited range of values, it can be useful for measuring websites according to their accessibility level. However, for many reasons the researcher has adjusted the metric to fit the third objective of this work which is to measure the degree of accessibility level improvement for a set of websites after fixing the top severe barriers detected by the metric. First, Zeng's metric needs a human expert to check the potential violations. Theoretically, this work's metric (WABS) can be used to calculate barriers weight based on all the success criteria in all WCAG 2.0 priorities. However, this research focuses only on the success criteria that can be evaluated using automated tools. Thus, the potential parameters that need human judgment will be ignored. Second, Zeng uses the total number of webpages checked in a website as a parameter. Nevertheless, this does not apply to this work's scope since only one webpage from each website will be checked which is the homepage.

Zeng's metric is analyzed by WCAG 1.0 checkpoints where Priority "A" violations weigh 3 times more than a Priority "AAA" violation, since Priority "A" violations pose more difficulties in accessibility than Priority "AAA" violations. Higher score means more accessibility barriers to the users with disabilities. Lower score indicates better conformance to the Web accessibility guidelines. A score of zero means the website does not violate any Web accessibility guidelines and must have no accessibility barrier to users with disabilities. Since this work focuses on WCAG 2.0, priority weights suggested by (Vigo et al., 2007) are utilized by WABS where (priority "A"=0.8, priority "AA"=0.16, priority "AAA"=0.04). Thus, the same priorities' weights will be used in the modified version of Zeng's metric. Formula (3.2) shows the modified version of Zeng's metric.

$$Web\ Accessibility\ Level = \frac{1}{NP} \sum_p (v * Wv) \quad (3.2)$$

Where,

P : the priority level that has been violated

v : The total number of violations (barriers) that violate the same priority level in a website

Wv : is the weight of the priority level that has been violated.

However, since only one webpage from each website will be checked, NP will be assumed 1, which leads to formula (3.3).

$$Web\ Accessibility\ Level = \sum_p (v * Wv) \quad (3.3)$$

To calculate the accessibility level for a website, for each violated priority level within this website, the total number of barriers that violates this priority level is multiplied by the priority's weight, and then the results for all priorities are summed up. Table 3.2 shows a simple hypothetical example on how the modified metric works on a dataset of three websites. The first website has 20 barriers (violations), where 15 of them violate priority A, 3 barriers violate priority AA and the rest 2 barriers violate priority AAA. The second website contains 10 barriers, where one of them violates priority A, 5 violate priority AA and 4 barriers violate priority AAA. The third website includes 15 barriers,

where 6 of them fall within priority A, priority AA has 4 while priority AAA has 5 barriers. The last row of the table shows the final results of each website. Website 1 scored the highest value (4.88) which means it is the least accessible one, while Website 2 is the most accessible website with the least score which is 4.32.

TABLE 3.2: An Example of How Zeng (2004) Modified Metric Works

Website	Website 1 (20 barriers)	Website 2 (10 barriers)	Website 3 (15 barriers)
Total number of barriers that violate priority A	15	1	6
Total number of barriers that violate priority AA	3	5	4
Total number of barriers that violate priority AAA	2	4	5
(V*W _v) for priority A	15* 0.16= 2.4	1* 0.16= 0.16	6*0.16= 0.96
(V*W _v) for priority AA	3* 0.8=2.4	5*0.8= 4	4 *0.8= 3.2
(V*W _v) for priority AAA	2*0.04=0.08	4*0.04= 0.16	5*0.04= 0.2
$\sum_p (v * W_v)$	4.88 (2.4 +2.4+ 0.08) score; least accessible)	4.32 (0.16+ 4+ 0.16) (lowest score; most accessible)	4.36 (0.96+ 3.2+ 0.2)

3.2.2.1 Research approach for measuring web accessibility level improvement

The automated tool has to evaluate a dataset of websites and generates for each website a separate report that includes the success criteria that have been violated along with the number of times a success criterion has been violated (frequencies). Based on the evaluation tool results the accessibility score will be measured for each website using formula (3.3). Then, WABS metric will calculate barriers severity for the evaluated dataset and rank the barriers accordingly. WABS does not rank the identified barriers for each website separately; instead it works on all websites reports generated by the automated tool to determine the common barriers within the whole dataset and then rank barriers based on their severities. Following, the highly ranked severe barriers detected by WABS will be removed from the evaluated dataset (since the researcher is not authorized to fix them). Once the top ranked severe barriers are removed, the accessibility score of the websites will be measured again using formula (3.3). Then the accessibility score of websites will be compared statistically before and after removing highly ranked barriers. This will allow answering the third question of this study which is: To what extent does fixing highly ranked severe barriers (detected by the proposed metric) would increase the accessibility level for a set of websites?

3.3 DATA COLLECTION PROCEDURE

Before conducting the experiments, the evaluation dataset has to be selected. The dataset contains the homepages of websites that will be evaluated. Two evaluation datasets are formed to be deployed within the experiments, namely set (A) and set (B). The datasets are selected from Alexa (2014) which shows the top high-traffic websites. Many large

scale accessibility studies such as (Sullivan & Matson; 2000; Zeng, 2004; Loureiro et al., 2015) selected their samples from Alexa since it reflects the popularity of a website. This enables to check whether the popular websites provide equal access and equal opportunities to all users regardless their disabilities. Alexa obtains Web traffic statistics by logging (anonymously, via browser plug-in) the sites that Alexa users browse. Traffic rankings are based on the number of times users have accessed a site in the last six months.

Set (A) contains 500 homepages of high-traffic (most popular) Malaysian websites, while set (B) contains 500 homepages of high-traffic global websites. All of the selected websites were in the Top Level Domain. The homepage was only checked since it serves as users' primary gateway to its entire subordinate content. The experiments will be carried out by evaluating these datasets using automated tools. Three experiments will be conducted on Malaysian websites, whereas the fourth experiment will be conducted on global websites to check whether the barriers weight significantly change when data are collected from different sample other than Malaysian websites. The full list of URLs can be found in Appendix B.

3.4 AUTOMATED TOOLS SELECTION

Despite the drawbacks of evaluating with automated tools, conducting experiments with large volumes of data is hard and tedious without their help. The automated tools help the Web designers and developers to assess the accessibility of the underlying code of websites by checking the code against guidelines and highlighting the barriers. Hence, the appropriate automated tools must be carefully selected in order to evaluate dataset (A)

and (B). There are many available automated tools to check websites' accessibility. Moreover, choosing the right tool is not a trivial task. Many factors need to be considered in order to select the appropriate tool to evaluate these datasets. Hereunder, seven factors suggested by (W3C (b), 2005; Groves (b), 2013) to be considered:

- A) **Guidelines:** Tools that support WCAG 2.0 will be considered. Some automated tools only support WCAG 1.0 and other guidelines, such as BITV and Section 508 .
- B) **Language:** The automated tool should support the English language. Otherwise, the work could be hard for the evaluator who does not speak that language.
- C) **License:** Open source and free software will be chosen instead of commercial off-the-shelf (COTS) products. Web developers and evaluators who run their own websites may try to avoid COTS due to a limited budget.
- D) **Ease of use:** Tools that are not Web browser plug-ins and operate by typing URL will be considered. These automated tools should be all web-based without the need to download or install anything else in order to use them (i.e. online checking).
- E) **Convenient testing:** Tools that provide paste option for website address instead of typing it would be more practical and convenient when testing large dataset.
- F) **Report's format:** automated tools must provide flexibility to the users by offering them various ways to view the generated reports based on their preferences, such as: HTML, PDF, XML and CSS.
- G) **Reports' presentation:** Automated tools are different in the way they present reports. Some tools present reports in a way that make it difficult to run statistical

analysis due to the layout style. When implementing WABS metric, it is essential to find a tool that makes it easy to calculate barriers' frequency.

Based on factors mentioned above, ten different tools were considered as listed below and were compared based on their conformance to the factors as shown in Table 3.3.

- 1) Accessibility valet (<http://valet.webthing.com/access/url.html>)
- 2) Accessibility Bookmarklets (<http://accessibility-bookmarklets.org/>)
- 3) BoIA's A11Y (<http://www.boia.org/Page/Accessibility/22>)
- 4) WAVE (<http://wave.webaim.org>)
- 5) Hera-FFX (<http://www.sidar.org/recur/aplica/heraffx.php>)
- 6) Vamola (http://www.validatore.it/vamola_validator/checker/index.php)
- 7) EvalAccess 2.0 (<http://sipt07.si.ehu.es/evalaccess2/servlet/Evaluate>)
- 8) aViewer (<https://www.paciellogroup.com/resources/aviewer/>)
- 9) A-Checker (<http://achecker.ca/checker/>)
- 10) TAW (<http://www.tawdis.net/ingles.html?lang=en>)

TABLE 3.3: Automated Tools Comparison

Tool Name	WCAG 2.0	English	Free	Online checking	Paste option	Different reports format	Well- structured layout
Valet	X	✓	✓	✓	X	✓	X
Bookmarklets	✓	✓	✓	X	✓	X	X
BolA's A11Y	✓	✓	X	✓	X	✓	✓
WAVE	✓	✓	✓	✓	✓	X	X
Hera-FFX	X	X	✓	✓	X	X	X
Vamola	✓	✓	✓	✓	X	X	X
EvalAccess2.0	X	✓	✓	✓	X	X	X
aViewer	✓	✓	✓	X	X	X	X
A-Checker	✓	✓	✓	✓	✓	✓	✓
TAW	✓	✓	✓	✓	✓	✓	✓

3.4.1 Description of Automated Tools

Based on the comparison in Table 3.3, two automated tools have found to cover all the factors, namely TAW and A-Checker. Thus, they have been selected to conduct the experiments. Table 3.4 displays twenty success criteria that represent the union of barriers that can be detected by both tools along with their description (Note that the barrier id corresponds to SC id that has been violated). The description of the full list of SCs which are not covered by the tools can be found in Appendix A.

TABLE 3.4: Success Criteria that can be checked by TAW and A-Checker along with their description

Guideline Number	Success Criterion ID	Success Criterion Description	Priority level
1.1	1.1.1	Alternatives for non-text content	A
1.3	1.3.1	Info, structure and relationships conveyed through presentation can be programmatically determined	A
1.4	1.4.1	Color is not the only identifier	A
	1.4.4	text can be resized without assistive technology up to 200% without loss of content or functionality except for captions and images	AA
	1.4.6	The visual presentation of text and images of text has a contrast ratio of at least 7:1	AAA
2.1	2.1.1	All functionality of the content can be navigated by keyboard only	A
	2.1.3	Whole website is accessible by keyboard	AAA
2.2	2.2.1	Time limits have timing controls	A
	2.2.2	Pause, Stop, Hide: For moving controls > 5 sec	A
	2.2.4	Interruptions can be postponed or suppressed by the user	AAA
2.4	2.4.1	Provide a “skip to content” link	A
	2.4.2	Web pages have titles that describe topic or purpose	A
	2.4.4	Link Purpose (In Context) : The purpose of each link can be determined from the link text alone	A
	2.4.6	Headings and labels describe topic or purpose	AA
	2.4.9	Link purpose is clear from its text	AAA
	2.4.10	Section headings are used to organize the content	AAA
3.1	3.1.1	The default human language of each Web page can be programmatically determined	A
3.2	3.2.2	On Input: Changing the setting of any user interface component does not automatically cause a change of context	A
	3.2.5	Changes of context are initiated only by user request such as slide show	AAA
3.3	3.3.2	Labels or instructions are provided when content requires user input	A
4.1	4.1.1	When parsing: No major HTML errors	A
	4.1.2	Name, role and value; all elements comply with standards	A

3.4.1.1 TAW – Test de Accesibilidad Web

TAW (TAW, 2015) was developed by the CTIC Centro Tecnológico (Spain). It is available as: standalone multiplatform software, web-based application for checking online and as Firefox add-on. TAW uses WCAG 1.0, 2.0 and its own set of heuristics for

mobile accessibility as a reference. This tool supports three languages, namely English, Spanish and Catalan. TAW allows the user to select the priority level he/she wants to meet then submit the webpage URL for evaluation. The discovered accessibility violations (barriers) are listed along with their frequencies and priority level. Table 3.5 shows barriers that can be evaluated automatically by TAW without human intervention. For simplicity, barrier id corresponds to SC id that has been violated. (Appendix C shows TAW interface as well as the layout of report generated by TAW). Although the tool allows the user to check against any priority level, barriers which violate priority “AA” cannot be detected by TAW as shown in Table 3.5.

TABLE 3.5: Barriers that can be discovered by TAW with the corresponding violated guideline and priority level

Guideline Number	Barrier ID	Priority
1.1	1.1.1	A
1.3	1.3.1	A
2.1	2.1.3	AAA
2.2	2.2.1	A
2.2	2.2.2	A
2.2	2.2.4	AAA
2.4	2.4.2	A
2.4	2.4.4	A
2.4	2.4.9	AAA
2.4	2.4.10	AAA
3.1	3.1.1	A
3.2	3.2.2	A
3.2	3.2.5	AAA
3.3	3.3.2	A
4.1	4.1.1	A
4.1	4.1.2	A

3.4.1.2 A-Checker – Accessibility Checker

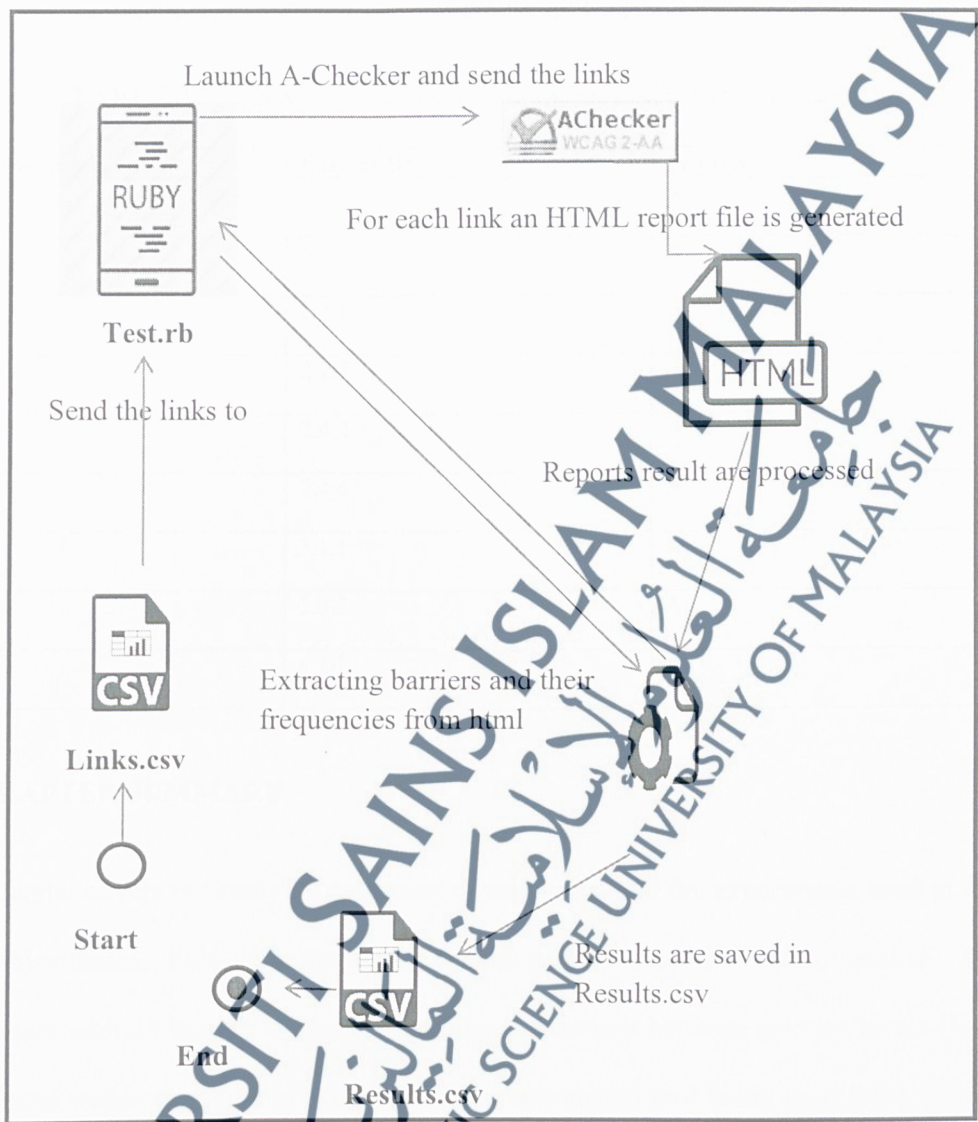
Accessibility Checker (A-Checker, 2014) is an open source accessibility evaluation tool that was developed in 2009 by the Inclusive Design Research Centre (formerly known as the Adaptive Technology Resource Centre) of the University of Toronto. A-Checker supports different languages such as English, German and Italian. The barriers that A-Checker can evaluate are listed in Table 3.6. The user then can select which guidelines to

evaluate against it, namely the HTML Validator, BITV, Section 508, Stanca Act, WCAG 1.0 and WCAG 2.0. Then he/she submits a webpage via its URL or by uploading its HTML file. The user can choose the report format based on the preferences such as: PDF, EARL, CSV or HTML. Unlike TAW, A-Checker reports show an (x) notation next to the violated SC without indicating how many times the error appears within the tested webpage. Therefore, the frequency of each barrier must be counted either manually or by automating the process (A-Checker interface and report layout can be found in Appendix D). To that end, a program was designed to count barriers frequencies automatically to reduce the time needed for analyzing the results. The program was written in Ruby language and can run on any platform that supports Ruby 2.0 and above. Ruby is a dynamic, reflective, object-oriented, general-purpose programming language that supports multiple programming paradigms (Wikibooks, 2015). Figure 3.3 shows the work flow of the program. The user must list the webpages he/she would like to test in a CSV file (links.csv). The program reads the links from links.csv and launches A-Checker. Subsequently, the program starts sending links one by one to A-Checker to be evaluated. For each link, an HTML report is generated by A-Checker. The program then parses the HTML report and extracts the frequency of each barrier. The results for each link are then saved in another CSV file named Results.csv. Once the results are saved in results.csv, they are ready for any further statistical analysis. (The source code can be found in Appendix E)

TABLE 3.6: Barriers that can be discovered by A-Checker with the corresponding violated guideline and priority level

Guideline Number	Barrier ID	Priority
1.1	1.1.1	A
1.3	1.3.1	A
1.4	1.4.1	A
1.4	1.4.4	AA
1.4	1.4.6	AAA
2.1	2.1.1	A
2.2	2.2.1	A
2.2	2.2.2	A
2.4	2.4.1	A
2.4	2.4.2	A
2.4	2.4.4	A
2.4	2.4.6	AA
3.1	3.1.1	A
3.3	3.3.2	A
4.1	4.1.1	A

FIGURE 3.3: A-Checker Report' Analyser Program



3.4.1.3 Common barriers between automated tools

TAW tool and A-Checker, like any other tool; cover only a portion of WCAG 2.0 because the violations that can be detected automatically vary from tool to another. In order to validate hypothesis three [H3], barriers weight yielded by A-Checker and TAW

must be compared. Therefore, the common barriers that represent the intersection of violations that can be detected by both tools are found first as shown in Table 3.7.

TABLE 3.7: The Common Barriers between TAW and A-Checker

Guideline number	Barrier ID	Priority
1.1	1.1.1	A
1.3	1.3.1	A
2.2	2.2.1	A
2.2	2.2.2	A
2.4	2.4.2	A
2.4	2.4.4	A
3.1	3.1.1	A
3.3	3.3.2	A
4.1	4.1.1	A

3.5 CHAPTER SUMMARY

This chapter covers the study's hypotheses' development and the experiments used to test these hypotheses. Five hypotheses are developed to test the metric quality. The hypotheses address barriers' weights regarding differences between priority levels [H1], differences within the same priority level [H2], automated tool being used [H3], testing the sample over time [H4] and testing with different sample [H5]. The hypotheses are validated through four experiments. Within each experiment, a dataset of websites will be evaluated automatically in order to check each barrier type and frequency. Then, WABS metric will be applied to the results of each experiment to test the metric quality. Once the metric quality is checked, the accessibility levels for a set of websites that contain barriers

will be measured using Zeng's modified metric (formula 3.3). Then, WABS metric will determine barriers severity in that set and rank them accordingly. Following, highly ranked barriers identified by WABS will be removed from the websites set and their accessibility levels will be measured again. A comparison between web accessibility levels before and after fixing highly ranked severe barriers will be carried out to meet the third objective of this work. Two evaluation datasets are formed, namely set (A) and set (B). Set (A) contains 500 homepages of high-traffic Malaysian websites, while set (B) contains 500 homepages of high-traffic global websites. These datasets are evaluated using TAW and A-Checker tools. The factors that led for selecting these tools for the experiments are discussed. Furthermore, the description of TAW and A-Checker as well as the success criteria that can be checked using them are demonstrated.