

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter provides a review of literature and researches relevant to the study. First, an overview of disability types is presented. Second, the web accessibility challenges faced by Web development teams, which lead them to program a code that does not attend to accessibility requirements, are discussed. The third topic, presents web accessibility metrics used to measure accessibility level. Following, different approaches aim at increasing web accessibility level, are highlighted. Section 2.6 shows the intersection of these disparate topics, which form the theoretical framework of this work. Finally, the conceptual framework of this research work is demonstrated. The chapter concludes with a summary that presents the highlights of each topic area relates to literature review of this research.

2.2 DISABILITIES AND WEB ACCESSIBILITY

The use of the Internet has been rapidly spreading to most areas of human life. In many industrialized countries, electronic or mobile governmental services are provided in almost all sectors such as immigration, education, commerce, news, workplace interaction, health care, recreation, and entertainment. This would enable citizens and residents to easily and efficiently access different services without usual problems, i.e. long queues, much time and effort. However, it is imperative to provide accessible Web

services to the majority of people including those with certain disabilities (permanent or temporary) so as to secure equal access and opportunities for everybody.

Four types of disabilities (visual, auditory, cognitive, and motor) are identified by the Center for Disease Control (CDC, 2016), which are mainly relevant to web accessibility. Visual disabilities include blindness, color blindness, and low vision. Color blindness as well as low vision make it harder for users to read the information on certain websites since dark backgrounds, unusual or small fonts, and unclear images pose problems for people with these two visual disabilities. Users with audio disabilities such as deafness or a hearing impairment are impacted when websites use audio files of low quality recordings. Users with cognitive impairments (learning disabilities) include autism, ADHD, and dyslexia as exemplars. Those with cognitive impairments can have difficulty reading text or lack the full ability to identify links within a website. Motor impairments include people with cerebral palsy, multiple sclerosis, muscular dystrophy, rheumatoid arthritis, carpal tunnel, broken bones, or other conditions that cause tremors or loss of fine muscle control. Users with a motor disability often have difficulty using their hands to navigate websites. Given these limitations, disabled users can use a variety of assistive technologies to gain access to the Web. Miller (2006) gave an example related to screen reader software interaction with webpage graphics. “In order to identify these elements to a screen reader, your site must provide ALT text, language that is associated with non-text elements that provides contextual meaning in cases in which users cannot see the graphic” (p. 21-22). Because screen readers only read text and cannot interpret graphic images, the code underlying the Web design should be written with titles, headings, and text captions that are appropriate for each graphic. Table 2.1 presents examples of

assistive technologies for each of the four disability types, suggested by (Bradbard & Peters, 2010).

TABLE 2.1: Disabilities and Assistive Technologies

Visual Disability
Screen magnifiers enlarge a portion of the screen as the user moves about the screen. For straight text, users can magnify on screen by zooming
Screen reader software present graphics and text as speech
Speech recognition systems allow people to make inputs with their voice rather than by mouse or keyboard.
Speech synthesizers allow users to hear the information they put into the computer
Refreshable Braille displays provide tactile output of information on the computer screen. Lines from the screen are sent to a device where small rounded plastic or metal pins are raised to form Braille characters. The user reads the Braille letters with his or her fingers, and then, after a line is read, can refresh the display to read the next line.
Braille embossers transfer computer generated text into embossed Braille output using a special printer
Talking word processors use speech synthesizers to provide auditory feedback of what is typed.
Large-print word processors allow users to view everything in large text without added screen enlargement
Auditory Disability
Telecommunications Device for the Deaf (TDD) provides means to communicate over phone lines using text terminals.
Closed captioning provides text translation of spoken material on video media (e.g., distance learning or video conference).
ShowSounds is a standard that provides visual translation of sound information. It is available in Windows XP and Vista. In Vista it is called "Captions."
Light signaler alerts the user when the computer is emitting sounds such as indicating a new email message.
Cognitive Disability
Reading tools and learning disabilities programs include software and hardware designed to make text-based materials more accessible for people who have difficulty with reading. Options can include scanning, reformatting, navigating, or speaking text out loud.
Screen reader software used for visual disabilities is also effective for people with dyslexia.
Speech recognition software can be used by people who find creating written language difficult.
Software like spell and grammar checkers, writing organizers, time management, and prompts are useful for processing impairments.
Office technology such as email, automatic reminders, and timers can be used for people with memory related impairments.
Motor Disability
Alternate pointing devices enable users with limited or no arm and hand movement to control mouse movements. Examples include foot operated mice, sip-and-puff systems, trackballs, headmounted pointing devices, and eye-tracking systems.
On-screen keyboards provide the key functions of physical keyboard and are typically used with alternate pointing devices.
Predictive dictionaries speed typing by predicting words as the user types them and offer words for the user to choose among.
Speech recognition enables users to control user interface or enter text via speech
Keyboard enhancements enable single finger operation of multiple key combos, delay onset of key repeat, bouncekey delays, or onset of inadvertent key presses (users with tremors).

Source: Bradbard & Peters (2010)

2.3 WEB ACCESSIBILITY CHALLENGES

Despite the importance of web accessibility as a research topic, the majority of websites developed remain inaccessible or semi-accessible (Hashemian, 2011; Nahon et al., 2012, Villena et al., 2014). This is due to reasons to be discussed in this section. In addition, web accessibility faces several challenges such as resource allocation, established managerial practices, and time limitation (Trewin et al., 2010). As a matter of fact, new research findings (Avila et al., 2012; Power et al., 2012; Brown & Scott, 2015) suggest an imminent need to look beyond the "how to" question of designing websites in order to consider accessibility constraints, accessibility context and the role of professionals who normally are involved in website development, e.g. Web designers, Web developers, and quality assurance engineers. This will make websites more accessible to different types of Internet users. The following is a list of challenges related to web accessibility in developing countries adopted from (López et al., 2011; Baowaly & Bhuiyan, 2012; Vu et al., 2012, Brown & Scott, 2015):

- Lack of accessibility awareness when designing and implementing websites.
- Limited resources allocated to cover accessibility issues, both tangible and intangible issues.
- Scarcity of professionals who are familiar with accessibility evaluation tools.
- Dearth of appropriate structured accessibility manuals for Web developers and unavailability of web accessibility training courses.

These and some other challenges, like lack of motivation among Web developers, contribute to making websites inaccessible for the disabled. Furthermore, lack of

efficiency among novices (inexperienced) in the evaluation process may impact web accessibility (Brajnik et al., 2011; Vázquez, 2014), and Web designers and developers may lack the necessary knowledge to implement techniques that support accessibility. All these problems contribute to the existence of several barriers in web accessibility and stimulate interest in this research domain to investigate more the “why” question: “Despite the availability of different resources, why are the majority of online websites still inaccessible?”

In fact, many research studies on accessibility have been conducted to evaluate the accessibility conditions of public websites (Olaferre & Lazar, 2011; Nizar et al., 2013; Lujan-Mora et al., 2014; Akgül & Vatansever, 2016). Nevertheless, few of these studies critically analyzed accessibility challenges (Harper et al., 2008; Kubitschke et al., 2013). Therefore, this research stresses the vital role of web accessibility not only for disabled Internet users but also for websites developers as well as IT team in charge of building and maintaining websites. It is believed that web accessibility is directly related to an organization’s profit, legislation, and image within society (W3C (a), 2012). It can be seen as a way for an organization to demonstrate its commitment to provide equal opportunities to all users. This can also be seen as a sort of corporate social responsibility. A major benefit of web accessibility for organizations is the potential direct and indirect financial gains from increased access to their websites.

This section surveys and contrasts various web accessibility challenges raised by current scholars in the last decade that face Web developers during design and evaluation phase and categorizes web accessibility challenges as proposed by (Abuaddous et al., 2016):

- Challenges related to web accessibility standards and guidelines.
- Challenges arising during website's design and development: motivation, awareness and other issues related to IT professionals who participate in the development of websites.
- Challenges during web accessibility evaluation: Evaluation tools and users.

To insure that a website is accessible, Web developers and designers have to follow guidelines at the design phase, and then test their websites accessibility against the guidelines to insure they are barrier free before releasing (W3C, 2014). The first two challenges mentioned above are related since they face Web developers at the design phase of a website. However, a separate section is dedicated to each challenge to explore it in details. The latter challenge faces Web developers at the testing (evaluation) phase before releasing, which lead them to ignore testing or to overlook correcting detected barriers. However, this categorization is not restricted and can overlap since the problems can be related at certain levels. For example, guidelines challenges can face Web developers at the design phase as well as the testing phase. Also, problems related to guidelines can discourage Web developers from following them during the design. Another example is lack of training which is related to learn the guidelines for designing accessible websites and to practice on automated tools to test the websites. Thus, lack of training can be affected by guidelines challenges as well as automated tools problems.

2.3.1 Challenges Related to Accessibility Standards and Guidelines

The increasing number of national laws and policies addressing the accessibility of information and communication technologies (ICT), including the Web, resulted in many

different approaches in practice. Some of the laws and policies are based on establishing a human right to ICT. Many adhere to the approach in which any ICT purchased by government must be accessible, while others believe that any ICT sold in a given market must be accessible, among other approaches (Internet Society, 2012). While many countries such as France and Spain have developed their own accessibility legislations, some developing countries still do not have a specific item of legislation protecting the rights of people with disabilities (Sloan & Horton, 2014). These rights are bound in a more general equality act and differ in the level of definition of disabilities and in successfully accessing and using digital content, products and services. Besides the mandatory accessibility legislations enforced by many countries, many voluntary standards and guidelines were founded to support social inclusion of disabled people.

While W3C's Web Content Accessibility Guidelines (WCAG) is the most internationally adopted voluntary web accessibility standard (WCAG, 2012), many other guidelines and standards were established as well (e.g. Section 508, BITV and Fujitsu). Furthermore, there is sometimes one version of a guideline or standard at the national level, different versions at the provincial or state level, and yet different versions adopted by commercial, educational, and non-governmental organizations within the same country. This section focuses on WCAG 2.0 since it has been lately the most widely used Web standard in the literature (Santarosa et al., 2014; Villena et al., 2014; Gordon & Lujan-Mora, 2015; Medina et al., 2015).

The World Wide Web Consortium (W3C, 2016) is an international consortium that aims to develop Web standards. Its mission is pursued through making general guidelines that

will lead to Web standards. W3C's Web Accessibility Initiative (WAI), which is part of the W3C, focuses on enabling people with disabilities to create and interact with Web content. WAI promotes: a) the implementation of web accessibility guidelines in advanced tools, and b) the improvement of accessibility evaluation tools (WAI, 2011). Together, these may increase the number of disabled people who use the internet.

The WAI has developed: 1) Web Content Accessibility Guidelines (WCAG) which describe how to make Web content and Web sites accessible, 2) Authoring Tool Accessibility Guidelines (ATAG) for the Web authoring tools used to create the content, and 3) User Agent Accessibility Guidelines (UAAG) for the tools used to access that content (e.g. browsers and media players). Despite the importance of ATAG and UAAG, this work will concentrate on WCAG challenges for two reasons: first, there is enough literature about the WCAG that can allow the researcher to extensively analyze it. Second, many automated tools that have been developed to perform website evaluation are using rules/criteria presented within WCAG guidelines. Within WAI, the WCAG documents (WCAG 1.0 and WCAG 2.0) were developed. The WCAG 1.0 was published and became a W3C recommendation in 1999 and was then superseded by WCAG 2.0 in 2008. Although it is possible to conform either to WCAG 1.0 or to WCAG 2.0 (or both), the W3C recommends the use of WCAG 2.0 for new and updated content.

WCAG 2.0 is organized around four design principles that provide the foundation for web accessibility (perceivable, operable, understandable, and robust) (WCAG 2.0, 2008). Under these principles there are 12 guidelines. Each guideline has one or more testable success criteria (SCs). There are 61 SCs at levels A (lowest), AA (medium), and AAA

(highest). Each guideline and success criterion has its own techniques. Techniques are classified by WAI as informative, which means they are not required. The basis for determining conformance to WCAG 2.0 is the success criteria from the WCAG 2.0 standards not techniques. Techniques may be “sufficient”, “advisory” or “failures”. The former are reliable ways to achieve accessibility and following them will be enough to meet a SC, but they are not mandatory: failing in a test procedure for a “sufficient” technique does not necessarily mean that the success criterion has not been satisfied in some other way (WAI, 2014). Advisory techniques are suggestions which have a positive impact on accessibility, but might not be enough on their own to ensure a particular success criterion is met. They are generally divided into: HTML and XHTML, CSS, client-side scripting, server-side scripting, SMIL, plain text, ARIA and Flash techniques. Failures are techniques describing examples of where a SC would not be met, which results in accessibility barrier. Table 2.2 demonstrates the WCAG 2.0 conformance levels (WCAG 2.0 guidelines list can be found in Appendix A).

TABLE 2.2: WCAG 2.0 Conformance Levels

Conformance Level	A	AA	AAA
Explanation	All SCs of level A are satisfied. This is the “minimum standard” which a website must meet to be considered accessible for any significant disability groups.	All SCs of Level A and AA are satisfied. This is a “professional practice standard”, which a website should meet to be accessible to a broad range of disability groups.	All SCs (at all conformance levels) are satisfied. This is a “gold standard” of maximum accessibility which some websites may choose .

In order to prevent the obsolescence of WCAG against the fast evolution of technology, the WAI removed the technical aspects of accessibility from their guidelines and SCs. Technical information regarding how to implement Web content with existing Web

technologies is now provided in separate documents. However, in spite of all these changes, recent studies (Cooper et al., 2012; Power et al., 2012) have shown that many of the problems raised in WCAG 1.0 still persist in WCAG 2.0 and there have been little improvement in the level of web accessibility.

Although WAI pursues its mission through the creation of guidelines, these do not guarantee accessibility (Miller & Fels, 2013). There is still a lack of empirical evidence to demonstrate that conformance to WCAG 2.0 leads to more accessible websites for disabled users. A study conducted by Power et al. (2012) showed that conformance of a website to WCAG 2.0 Level A does not mean that disabled users will encounter fewer problems.

Many issues have been raised about WCAG 2.0, such as the level of understanding of accessibility issues required when using them (Bitar et al., 2012; Conway, 2014). One common challenge that is usually faced by Web developers is their inability to interpret or understand guidelines to enable accessibility. WCAG 2.0 documents are not easy to understand and require a certain level of technical knowledge of accessibility. Therefore, when developers or designers are required to implement accessibility, they do not always understand how to achieve the desired requirements. Alonso et al. (2010) showed that a group of 25 novice evaluators struggled to consistently rate problems according to WCAG 2.0. One cause of this was in the interpretation of the guidelines and the SCs. Besides the manual verification is needed by evaluators and Web developers, guidelines-based accessibility evaluation has several disadvantages (Tanaka & Rocha, 2011). WCAG 2.0 documents are difficult to navigate and locate. All the documents related to

these Guidelines exceed 450 pages with few hundred navigation links on each single page. Moreover, WCAG 2.0 tends to create supporting documents that can be updated more regularly than the standard itself, which can be a burden to Web developers since exploring these updates, necessitates both time and effort. As such, WCAG 2.0 could be unusable by real-world developers. In addition, WCAG 2.0 development itself is inaccessible to anyone who does not understand English.

Loitsch et al. (2013) pointed out that there exist ambiguities in the language used in WCAG 2.0. For instance, WCAG 2.0 stated that all “the success criteria (SCs) are testable when people who understand WCAG 2.0 test the same content using the same SCs, the same results should be obtained with high inter-rater reliability”. However, no explanations were provided about the minimum number of testers needed to fulfill the required “high inter-rater reliability”; also, the level of agreement between evaluators is undefined. Should all the testers obtain the same results or a certain portion of them is enough? Furthermore, Loitsch et al. (2013) mentioned the need for people who understand WCAG 2.0 tests without clarifying the level of expertise needed. This can lead to many different interpretations of what is required. This ambiguity could affect the efficiency and the effectiveness of the evaluation process. Eventually, this may create a gap between evaluators and the evaluation process.

Petrie et al. (2011) conducted interviews with 14 web accessibility evaluators. They found that the evaluators were unclear on the differences between automated and manual testing of accessibility and what guidelines can be tested through the automated tools. In addition, Brajnik et al. (2010) had 22 expert and 27 non-expert evaluators to perform

accessibility evaluation using WCAG 2.0 and discovered that 50% of testers were unable to come to an 80% level of agreement about whether a problem was present in a webpage. Additionally, when using WCAG 2.0 20% of the problems reported by the evaluators were false positives and 32% of the true accessibility problems had been missed. A false positive occurs if the violation has not taken place and was reported, whereas a false negative occurs if the violation really has taken place and was not reported. For the non-experts, the results were even worse having higher levels of false positives.

The focus of WCAG 2.0 is on the technical artifact, e.g. the webpage - not on users and their goals. This means that the activity of WCAG conformance is oriented towards testing these technical artifacts against SCs - rather than evaluating the user experience with specific impairments trying to complete specific tasks. Although a technical testing's focus can be helpful for programmers treating accessibility evaluation as a bug-fixing activity, this level of technical focus inevitably creates a gap between accessibility and user experience for disabled people (Cooper et al., 2012). Of a greater concern is WCAG 2.0's emphasis on perfect score on all SCs for a level to get the level conformance logo, which makes it impossible for some websites to achieve any acceptable level of conformance. WAI ignored the fact that in website development process, developers do not seek perfection; instead, they aim for a continuous, pragmatic improvement over versions (Hassell, 2013).

2.3.2 Challenges during Website's Design and Development

The main issue of web accessibility is that many websites are not designed with equal access in mind (Bradbard & Peters, 2010). In other words, lack of web accessibility is more a result of faulty design rather than inadequate technologies. Carter & Markel (2001) estimate that one percent of Web developers take accessibility into account when designing web pages. When websites are designed and developed without concern for users with disabilities, barriers that inhibit access to the content of the site will exist. This happens when Web developers write a code that does not attends to accessibility guidelines. Beside the problems related to web accessibility guidelines which are discussed earlier, other issues can contribute to the faulty design before testing and evaluating accessibility, such as: lack of awareness and motivation and the scarcity of professional training courses that may handle accessibility issues (Nahon et al., 2012; Villena et al., 2014; Gordon & Lujan-Mora, 2015). As a result, Web developers do not follow guidelines at the design phase of a website.

Developing websites requires a team effort that includes developers, designers, content producers and project manager. For the Web community, each member of this team has certain responsibilities related to his skills in delivering an accessible website. However, it is the project manager who monitors high-level adherence to business goals, and without his support, there will not be any changes to accessibility unless governments truly start enforcing legislation. Without a managerial impact on accessibility, it is quite normal for a Web team to ignore this accessibility issue or pretend they have taken care of it (Pereira et al., 2014). A project manager should ensure the early engagement of his team in the accessibility process. He should motivate and encourage the Web team to get

involved in the accessibility issues at early stages. Project managers are responsible for showing the importance of web accessibility to their teams as well as the positive effect of web accessibility on the commercial and community levels. If none of the project members is experienced in accessibility, then a third party can be consulted or outsourced to perform this task. Lack of accessibility may result in the marginalization of certain user groups, preventing them from accessing the website.

This section discusses the main challenges faced in projects related to accessible website during the design and the development of a website. Initially, the section highlights the accessibility awareness and motivation among the project's members especially IT developers. Then, it focuses on the benefits of web accessibility training for Web development staff.

2.3.2.1 Lack of accessibility awareness and motivation

Challenges such as lack of awareness are very common among Web developers, besides motivation, knowledge and guidance. The latter challenges are faced by most novice evaluators. Whether facing experienced Web developers or novice evaluators, all challenges contribute to the continuing presence of accessibility barriers on websites.

There are moral, legal and economic arguments for implementing accessible websites promoted by advocacy organizations such as disability related charities and academic organizations (Yi & Kang, 2012). Nevertheless, few Web designers follow accessibility guidelines (Nganji et al., 2013). Although the WAI of the W3C have published online guidelines, most IT professionals are unaware of them, and companies that have government contracts are mandated by their government's legislation to make their

websites accessible (Brown & Scott, 2015). The low level of accessibility is likely to be the result of several factors. For instance, despite the increase on awareness pertaining to accessibility over the last few years at the level of government and legislation, Web designers' knowledge remains quite low (Power et al., 2012).

Pye (2011) tested the importance of accessibility awareness for IT professionals and its effect on the accessibility level for visually disabled users. The author used a questionnaire and interviewed members of charity organizations that concern themselves with the matters of web accessibility for visually disabled persons. The results of the study revealed that there are certain rules to increase awareness of all individuals involved in Web development. Country-specific legislation should be introduced in web accessibility. There is also a key requirement for the continuation of the other current enforcement methods, such as assistive technology, to overcome the accessibility barriers. However, no precise recommendations are given by the author to treat the highlighted awareness problem.

In the USA, Yi and Kang (2012) showed the lack of awareness in public libraries. These libraries usually do not consider community members with physical disabilities when designing their websites. Therefore, the findings suggested that public library websites are not suited to deliver effective information services for users who need special assistance. Furthermore, this study revealed that public libraries do not consider having an appropriate funding in their reserved budget to support accessibility.

In the research conducted by Hashemian (2011), twenty (20) websites of Finnish higher education's institutes have been examined to evaluate their web accessibility by an

application software named TAW (TAW, 2015). The results showed that 30% of tested websites have full automatic accessibility in priority 'A' level but none of the tested websites has 'AA' compliance. Most of the tested websites have low accessibility. The authors claim that the lack of accessibility awareness among developers is the main reason.

Until now, the majority of IT professionals, especially those who are involved in developing websites, are not motivated to learn more about accessibility. This is because they do not have a comprehensive knowledge of accessibility and the difference it makes to disabled users and community commercial organizations (Villena et al., 2014). So, the challenge is to increase the level of awareness of the IT Web team regarding the impact and rewards of web accessibility as well as to motivate them through training course that highlights these rewards and benefits (Baowaly & Bhuiyan, 2012). On the other side, the lack of motivation does not necessarily mean lack of awareness since many IT professionals are aware of web accessibility legislations but not motivated to follow them.

Rosson et al. (2005) conducted a survey involving 634 Web developers. The aim was to understand the needs, problems and the processes that developers follow and the tools they use in websites development. During website implementation, the authors have observed that while developers were quite conscious of the overall quality and usability issues, only 5% of the respondents performed web accessibility testing. They hypothesized that this may be partly due to a lack of knowledge and motivation because of the relatively tedious and time-consuming testing required. Although Rosson et al. (2005) experiment was conducted 11 years ago; the motivation factor has remained an

issue until now. One of the common justifications given by the developers to abandon accessibility according to Gordon and Lujan-Mora (2013) is the limited time or resources' incompatibility of technology to accomplish this task. This happens because the accessibility specialist is just a small cog in a big wheel or even there is no specialist at all in the organization.

Trewin et al. (2010) surveyed IBM rich Internet application developers with varying levels of accessibility expertise to explore two concerns: barriers faced by developers when designing accessible web-based products and the value gained from the accessibility test. The findings showed that even for developers with experience in web accessibility, testing has been seen the most time-consuming among other phases with relatively low impact on the website. Nahon et al. (2012) reported obstacles and incentives for non-professional Web makers (e.g. blog writers and/or creators of personal websites) to consider accessibility in their work. The authors presented a theoretical framework that described variables they hypothesized that would contribute to designers' intention. They found that intrinsic motivation was the strongest predictor of a positive attitude that affected the intention to make technology accessible.

2.3.2.2 Lack of training

No one can deny that professional training, good education and guidance are crucial to produce skilled IT staff members who can understand the urgent need for accessibility. The training should cover different aspects related to web accessibility such as training on: disability types and their interaction with the assistive tools, guidelines checking and accessibility evaluation using different techniques. The lack of skilled staff may eventually negatively impact accessibility. However, accessibility training for Web

developers is still not available in most IT training centers. Thus, many Web developers continue producing websites that are inaccessible.

Many developers claim that accessibility is difficult to accommodate. Part of the problem is a lack of exposure to accessibility during training (Lazar et al., 2013; Vázquez, 2014; Gordon & Lujan-Mora, 2015). Most Web designers and developers have little or no experience to ensure that their code meets accessibility requirements. Designers approach accessibility problems differently than other IT professionals, such as Human Computer Interaction (HCI) specialists and developers. Designers and developers typically tend to think that Web users are just like them (Krug, 2013).

Recently, developers have indicated a need for more education in accessibility. An exploratory study (Lopes et al. (a), 2010) on the current state of accessibility surveyed more than 400 developers throughout Europe. The study found that 85% of developers wanted more accessibility training with more information on guidelines, disabilities and the use of assistive devices. Moreover, Web testers need training in accessibility evaluation methods like conducting real user testing with blind users and using screening techniques besides assistive technologies. One Brazilian study (Freire et al., 2008) revealed that while 45% of web-related professionals were aware of screen readers for blind users, they did not know how to make webpages screen reader-compatible.

There are a great number of recommendations for accessible development, but these are often distant from the developers' way of programming. As a consequence, developers do not follow them (Freire et al., 2008; Bittar et al., 2012). Avila et al. (2012) stated that for Web developers to create functionally accessible web resources, more than general

guidelines and evaluation tools are required. Thus, it is recommended that more training be given to Web development teams in order to raise their awareness and give them a better understanding of end-users.

2.3.3 Challenges during Evaluation

Web accessibility evaluation methods have been widely studied, and different evaluation methods have been proposed, e.g. (Brajnik et al., 2010; Brajnik & Harper, 2013). Different AEMs lead to different kinds of results and quality. They require different levels of resources and differ in their applicability (Brajnik et al., 2010). These methods have contrasting advantages and disadvantages and various properties (Gonçalves et al., 2014). For example, they differ in many ways:

- Method of implementation (analytical or observational).
- Users performing the evaluation.
- Purpose of implementation.
- Methods used in obtaining results.
- Cost/resources spent in evaluation.

In the next subsection (2.3.3.1), challenges in the accessibility evaluation process faced by IT staff when they test websites using automated tools are covered. User testing, screening techniques and other assistive technologies problems are highlighted in subsection (2.3.3.2), since they are related to each another.

2.3.3.1 Automated testing problems

WCAG can be checked manually, though verifying a site's accessibility manually can be time-consuming. Thus, software developers have created a number of tools to simplify

this process. Web accessibility testing tools are software programs and Web based services that help in determining whether a website meets accessibility guidelines (Brajnik et al., 2010). These tools can help reduce the time and effort required for the evaluation process especially for large-scale studies. Testing normally involves an evaluator to check conformance of a webpage against the accessibility rules encoded in the tool. A-Checker (2014), WAVE (2015) and TAW (2015) are examples of automated evaluation tools. Automated tools vary in the way they apply accessibility standards or guidelines to produce a list of automatically detected problems (Lazar et al., 2013). Generally, expert users are required to follow up with the results derived by the tool to determine the rate of accessibility.

Common drawbacks of automatic accessibility tools have been highlighted by many scholars, e.g. (Lazar et al., 2013; Gordón & Moreno, 2014; Martínez et al., 2014). There is not much information about the difficulties that evaluators may face when assessing accessibility using these tools (Tanaka & Rocha, 2011). However, one of the most known deficiencies of automated tools is the difficulty in interpreting results. At first glance, running evaluation using automated tools sounds easy. Unfortunately, they are time consuming when the novice Web developers try to analyze their results. Moreover, interpreting these results requires an expert Web developer with a technical background in web accessibility. This necessitates constant checking against manuals and documentations. When these tools are used by practitioners who lack experience or knowledge in accessibility, the results can be unpredictable and the quality of findings questionable.

Automated testing covers only a small proportion of WCAG 2.0 and is unable to check every aspect of accessibility (Lopes et al.(b), 2010), but it is possibly able to flag items that need to be manually checked. Some tools like TAW do not evaluate the Web content using all guidelines criteria and the rest are usually evaluated by testers who are commonly left to their own interpretation of those guidelines (Avila et al., 2012). Moreover, there is high detection rate of defects among many of evaluation tools, which requires a user's manual inspection and, thus, resources are wasted (Johari & Kaur, 2012).

Trewin et al. (2010) interviewed 49 IBM Web developers and revealed that the majority of the existing evaluation tools are often unclear, cumbersome and incomplete with respect to standards or guidelines that must be met. Moreover, some automated testing tools do not provide support for changed or newly developed accessibility guidelines. While tools can check the adherence to a number of guidelines, some guidelines are not checked properly. For example, it is hard for tools to ascertain whether the "alt" attribute is meaningful or not. This can result in high false positive and inaccurate results. Inaccurate results can decrease the efficiency and increase the cost of accessibility testing. Trewin et al. (2010) showed a need for new methodologies, particularly mixed ones that may reduce false positive results. Precisely, to ease end-user interpretability, light-weight visualizations that support novices in performing manual checks and simplification of the automated tools' results are needed. The research community of accessibility is still unable to standardize ways or methods that are fully automated to generate accessibility rate for a specific website. Developers of interactive systems, especially websites, are frequently not specialists in accessibility and usability techniques. In the absence of an

accessibility expert, the evaluation tool might go a long way to find adherence to the standard guideline. But even hiring an accessibility expert may often be beyond the financial capabilities of a typical website development project.

Two experiments were conducted in 2012 and 2013 by (Abuaddous et al., 2013) using the following tools: Accessibility Check, A-Checker and TAW for 20 public universities in Malaysia in order to diagnose the overall accessibility of these universities' websites. Unfortunately, none of the websites were fully accessible based on the results obtained by the selected tools. In addition, the results obtained varied and were expressed in different numbers of errors due to the fact that accessibility tools differ in their interpretation of the WCAG, which supports contention (Al-Khalifa et al., 2012). The authors suggested that automated evaluation tools may underestimate or overestimate the number of accessibility errors on a webpage, making the reliability of such tools doubtful.

Choosing website accessibility evaluation tools is not a trivial decision: whether or not to pick the easiest one to use, or the fastest one to learn or the one which produces low false positives. The choice of the tools without previous experience can be a daunting task, especially when working under the pressure of time constraints (Greeff & Kotzé, 2009). Many factors need to be taken into consideration when choosing the suitable evaluation tool. The tool should be clear and user-friendly, provide high quality and reliable results and be capable of testing the browser's DOM in order to test what users are experiencing. Additionally, the tool must be suitable to integrate in the development process and flexible enough to be used in any environment by any team in any location. While many tools provide some guidance on verifying results which need to be manually checked, few

of them provide easy to follow and user-friendly guidance on performing such manual inspection. There are other factors, including but not limited to the ones mentioned above, to consider when choosing the appropriate evaluation tool (Groves (a), 2013). Section 3.4 in Chapter 3 highlights the factors to be considered when selecting automated tools for the study experiments.

2.3.3.2 Users' testing problems

Although accessibility tools and guidelines can help in detecting accessibility problems, user testing should be applied whenever it is possible to verify the results (Theofanis & Redish, 2003). User testing is a process where formal or informal experiments are set up with real users who are asked individually to perform goal-free or oriented navigation on a website, and whose behavior is observed by experienced evaluators. Lab studies with real users are the most effective when conducted early in the lifecycle of a product but can also be conducted towards the end of the iterative design cycle (i.e. when large accessibility problems might be ignored to ship a product or release a site on time). Involving users with disabilities in evaluation has many benefits; however, it cannot alone determine if a website is accessible or not (Horton & Quesenbery, 2014).

Several studies of usability evaluation methods (Nielsen, 2000; Brajnik et al., 2010) have shown that user testing methods may fail in yielding consistent results when performed by different evaluators and that an inspection-based method is not shortcoming free. Other difficulties were observed, like preparing testing materials, developing realistic scenarios of tasks and choosing the right sample size for the test (Henry, 2007). In addition, this kind of testing has been criticized for being subjective and for depending on

the users' background as well as the Internet skills (Aizpurua et al., 2014). Although accessibility and usability are two different subjects, there is no reason to assume that the kind of uncertainty that applies to usability evaluation methods should not apply to accessibility as well.

User testing is a part of usability testing, but user testing with disabled users also adds accessibility findings to the normal usability findings. User testing can be conducted in labs or remotely (Madathil et al., 2011). In lab testing, a live moderator or observer prepares a scenario and uses a think-aloud technique with high facilitator interaction. Data collection and tasks would focus on specific areas of concern for potential accessibility problems, rather than general site usage. On the other hand, in remote user testing, the tester moderates the test via communication means (e.g. phone, webcam, and web-based tool) rather than being on-site with the user. This is often done due to budget constraints in setting up appropriate labs or when the targeted users are scattered across the country.

Recruiting qualified disabled users is not a trifling task since there is a complexity associated with engaging the right users who match the website audience demographics (Nielsen, 2012). People with disabilities are diverse and use different interaction techniques and assistive technology. It is important to recruit people with disabilities, characteristics and expertise depending on the target audience. However, accessibility teams often have limited access during testing phase to real disabled users that meet the end-user profile and who can take time out to do the evaluation. Users regularly have inadequate means to attend to the evaluation. The use of varied assistive devices, like

screening techniques, may cause further problems during the evaluation, especially due to the fact that the mastery of a particular device does not mean an automatic skills transfer to another or newer device, besides hardware incompatibility (Greeff & Kotze, 2009). Thus, recruiting users for testing normally requires that the level of technical savvy of the tester is known and documented (Brajnik et al., 2011). Novice moderators with limited experience can get confused by the type of difficulties being faced by users. It is important to differentiate between challenges in dealing with assistive technology (especially when the assistive tool is a different product than the user is familiar with) and challenges being faced with the website being tested.

User testing with small populations often provides useful information, and normally when combined with more qualitative techniques can help to understand users' behaviors. However, it is not statistically robust due to the small participants numbers (W3C, 2010). It is hard to draw conclusions from limited studies, and results cannot be generalized to all people with similar disabilities. User testing with special populations can be beyond the expertise or financial resources of a typical Web developer and can incur greater time and monetary costs due to special arrangements for testing (Nielsen, 2012).

It is vital to know the type of adaptive strategies or assistive technology being used by users to arrange a lab that simulates their real interaction experience. Assistive technologies are software and hardware that people with disabilities use to improve interaction with products. In the case of web accessibility testing, it is preferable to ask users to bring assistive technology they are familiar with to make them feel comfortable. Unfortunately, it is not always the case that users are being able to bring their assistive

technologies. Many assistive technologies being used by real users can be very expensive and demo versions are limited which make the cost of preparing appropriate labs high. Time requirement escalates with the need to prepare labs equipped with appropriate assistive tools. This kind of procedure can be beyond the testing cost of the IT team. In some cases, it might be best to go to the user's place, rather than having the user come to labs. A drawback to this is that limited numbers of the project team get direct interaction with the user.

Generally, remote testing is less expensive than a traditional lab testing (Bruun & Stage, 2012). Nevertheless, it embeds many challenges like users being unable to share their screen over the Internet or having unreliable or slow connection speeds. Moreover, there is the restriction of the user's body language that might inhibit some of the cues to their reactions to the website being tested.

2.3.4 Web Accessibility Challenges Summary

The previous web accessibility challenges are presented in a table format along with their various associated forms as depicted in Table 2.3.

TABLE 2.3: Web Accessibility Challenges Summary

Challenge Forms	Examples
Standards and Guidelines Challenges	
Ambiguity	-The WCAG 2.0 includes some measurements that need to be quantified in order to be interpreted in the same way by all Web developers
Only English version is available	-Inaccessible to developers who do not understand English Language
Incompleteness	-The application of WCAG 2.0 alone is not sufficient to guarantee website accessibility
Hard to navigate	-The guidelines normally are presented in descriptive texts, navigation links and tables, which is not an effective way of display for the designers and developers.
Not suitable for naïve web developers	-Following WCAG 2.0 requires qualified Web developers with certain level of technical knowledge of accessibility in order to read and interpret the guidelines.
Requires perfection	-All the SCs on each level have to be met in order to get the conformance logo which could be frustrating to Web developers
Inefficiency	-Following the guidelines could make the accessibility evaluation process slow. Furthermore, even in large companies, resources including time, money and staff are hard to allocate for accessibility.
Challenges during Website's Design and Development	
Lack of Accessibility Awareness	-Despite awareness about accessibility over the last few years at the levels of government and legislation, Web developers are unaware of the legal and industry requirements of effective accessibility and their level of knowledge remains quite low. -Despite the availability of WAI guidelines online, most IT professionals are unaware of them and do not use them.
Lack of Accessibility Motivation	-Many Web developers do not have a comprehensive knowledge of accessibility and the difference it makes to disabled users. Thus they are not motivated to learn about guidelines. -Learning guidelines and how to evaluate against them can make testing process slow which unmotivated Web developers and designers to follow them.
Untrained IT Team	-lack of exposure to accessibility during training courses for Information Systems professionals -Most Web designers and developers have little experience to ensure that their code attends to accessibility requirements.
Challenges during Evaluation	
Automated Tools Problems	-Choosing the right accessibility tool can be time consuming. -The need for manual inspection along the tools. -Some guidelines are not checked properly (e.g. "alt" attribute). -Automated tools results are hard to analyze by inexperienced Web developers, which can affect the quality of the test process. -Can give inaccurate results and results can vary from one tool to another, depending on interpretation of the guidelines.
User Based Testing Problems	-The cost of setting up labs is quite high. -User testing with large samples is often beyond the expertise or financial resources of a typical Web developer and is more time consuming than other methods. -Finding disabilities that match the website audience can be difficult. -The need for expert evaluator -Difficulties in preparing testing materials, in developing realistic scenarios of tasks and in choosing the right sample size for the test

2.4 MEASURING WEB ACCESSIBILITY LEVEL USING METRICS

The idea of developing web accessibility metric is not new as many previous studies proposed quantitative metrics to measure websites accessibility. (Sullivan & Matson 2000; Zeng, 2004; Arrue et al., 2005; Velleman et al., 2006) defined metrics all based on failure rate.

Sullivan & Matson (2000) proposed a metric to classify websites based on their accessibility into four levels: highly accessible, mostly accessible, partly accessible and inaccessible. The websites have been evaluated automatically and manually to identify real and potential violations, then were classified based on failure rate (FR). The FR was defined on the basis of a subset of WCAG 1.0 checkpoints, where eight checkpoints were evaluated.

$$FR = \frac{\text{Real violations (checkpoints)}}{\text{Potential violations}}$$

The advantage of such a metric lies in its simplicity. However, it does not consider other factors such as error impact, error nature and severity of detected violations.

Zeng (2004) proposed the Web Accessibility Barrier (WAB) formula to evaluate websites accessibility, where a high WAB means a low accessibility level. This formula uses the total number of a website pages (NP), total accessibility errors, potential errors in a webpage and error priority as input parameters. Zeng's formula is analyzed by WCAG 1.0 checkpoints as:

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

Where, Wv is the weight of violations in inverse proportion to WCAG priority level. However, the returned ratings of this metric are neither restricted to a limited range of values nor are normalized. Moreover, this metric needs a human expert to check the potential violations.

The Web Accessibility Quantitative Metric (WAQM) was proposed by Arrue et al. (2005). This metric is calculated automatically from evaluation reports yielded by the automated tool when checking against WCAG 1.0 checkpoints. WAQM considers the failure rate of each checkpoint (c) on each page (p), $FR(p, c)$. The failure rate is transformed through a piecewise linear function that approximates a hyperbole to values that are more spread out as they get close to 0. These values are then weighted by priority of the checkpoint, and finally weighted by the following formula:

$$d_p = e^{-i}$$

Where (p) is a page and (i) is its depth level in number of links from the home page (for which $i = 0$). Weights associated to priorities, slopes and intercepts of the piecewise linear function need to be experimentally tuned. Additional study (Vigo et al., 2007) discussed the dependence of the WAQM metric on the tool being used and shows that although the numeric values produced by WAQM are tool-dependent, the ranking of webpages and web sites does not depend significantly on the tool. Despite that WAQM has advantage of producing normalized quantitative values depends on checkpoint priorities, it does not

consider the error rate of the tool being used and it does not provide means to combine automatic and manual evaluations.

Velleman et al. (2006) proposed The Unified Web Evaluation Methodology (UWEM) which is a methodology designed to evaluate accessibility against WCAG 1.0 by integrating manual and automatic evaluation methods. The accessibility value of a page (p) is represented by the following formula:

$$AV_p = 1 - \prod_c (1 - R_{pc} F_c)$$

Where F_c is the probability that a violation of checkpoint (c) results in a failure mode (F_c is assumed to be constant, set to 0.05). R_{pc} is a factor that depends on the agent who performs the evaluation of the checkpoint. If the evaluation is carried out by a tool, then R_{pc} is the failure rate combined with the error rate of the tool; if it is an expert, then R_{pc} is the probability of error of the expert. AV_p gives the probability that the page results in a failure mode; the mean for overall webpages gives the accessibility value for the entire website. The advantages of this metric are that it produces a quantitative, normalized and absolute value and it was derived based on a clean mathematical background. Moreover, the metric considers error rates, the impact on users, and supports the integration of manual and automatic evaluation methods. However, no suggestions were given on how to estimate these parameters in a reliable manner. Bühler et al. (2006) proposed an extension of the UWEM in order to adapt measurement to different disability groups. Therefore aggregation model was proposed. The model is simplified as following:

$$A(u) = 1 - \prod_b (1 - R_b S_{ub})$$

Where R is the evaluation report and S is a severity value from 0 to 1 (for each barrier type b and user group u). However, this metric reliability might be doubtful since it was evaluated by a preliminary experiment which involved the assessment of only six websites conducted by six disabled users.

Brajnik & Lomuscio (2007) proposed SAMBA which is a method for measuring websites accessibility based on WCAG 1.0. This method uses the output of automated tools and couples it with human judgment, so that correct estimates of tool errors can be assessed, and that estimations of severities of barriers are used. On the other hand, MAMBO (Brajnik (b), 2008) is a method applied in a similar way, where automatic tools cannot be utilized. However, applying these methods absolutely needs human expert for evaluation, otherwise cannot be useful.

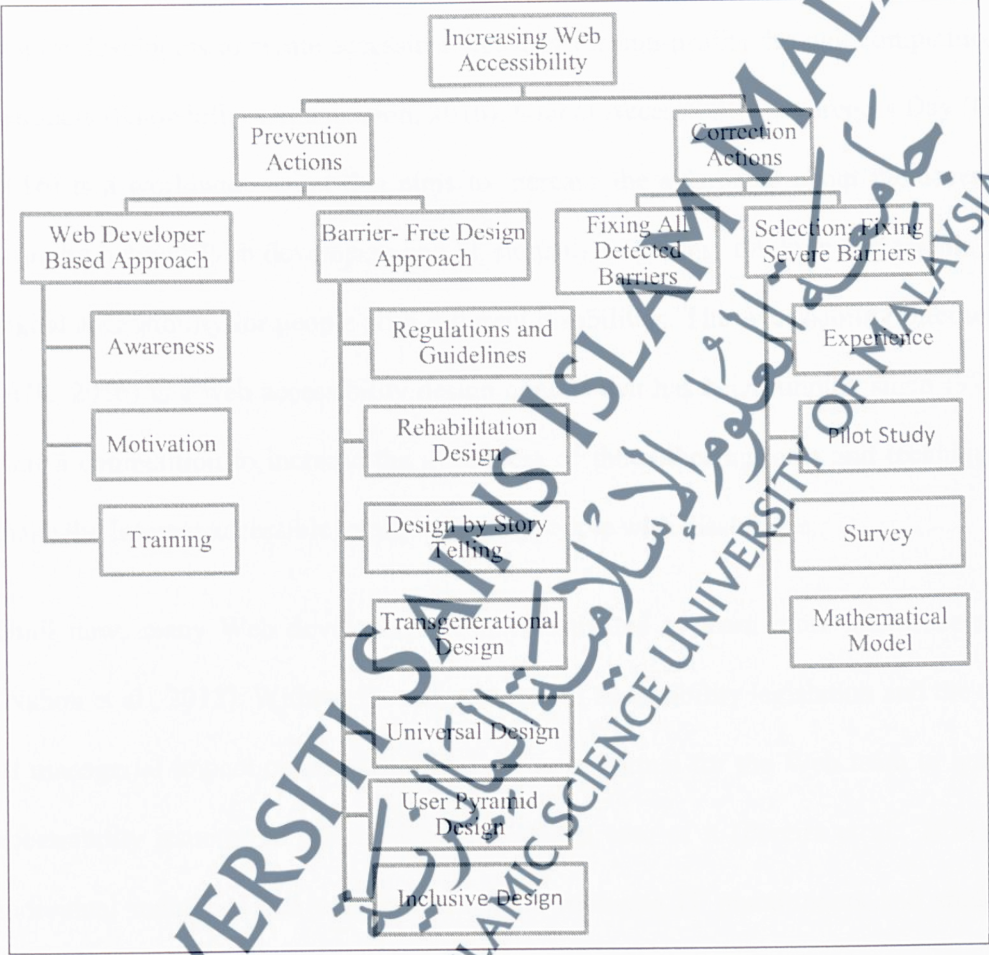
The previous conformance based metrics contributed to the web accessibility field by quantifying the accessibility measurements. Nevertheless, there are still some interesting and relevant problems to be addressed. There is an imminent need to look beyond the "Is the website accessible? and to which level?" and "What are the violated priority levels?" questions and investigate the commonplace barriers spread across the Web and contributing to low accessibility level.

2.5 APPROACHES TO INCREASE WEB ACCESSIBILITY LEVEL

This section of literature review outlines the approaches for increasing web accessibility level. It is divided into two main sections, namely Prevention Actions and Correction Actions. While the prevention actions discuss the actions and approaches to be followed

to prevent the occurrence of a barrier when designing a website, the correction actions deal with fixing barriers detected while testing and evaluating a website against accessibility guidelines (Robitaille, 2002; Stamatis, 2003). Figure 2.1 demonstrates the different approaches to increase the web accessibility.

FIGURE 2.1: Approaches to Increase Web Accessibility



2.5.1 Prevention Actions

This section explores the approaches and actions that improve the accessibility level when designing a website by preventing possible barriers. First, the section highlights Web developer based approach that relates to increasing awareness, motivation as well as

training. Then, it compares variety of barrier-free design approaches that emerged to facilitate accessibility.

2.5.1.1 Web developers based approach

Lack of awareness is very common among Web developers and when it comes to increasing awareness about web accessibility, the non-profit organization continues to inspire developers to create accessible websites for non-profits through competitions and seminars (Knowbility Organization, 2016). Global Accessibility Awareness Day (GAAD, 2016) is a worldwide event that aims to increase the awareness about the accessibility issue by helping Web developers and IT students in talking, thinking and learning about digital accessibility for people with different disabilities. The Accessibility Internet Rally (AIR, 2016) is a web accessibility design contest that has been running since 1998. AIR runs a competition to increase the awareness of the authoring tools and techniques that make the Internet accessible to all, including people with disabilities.

Until now, many Web developers are not motivated to learn more about accessibility (Nahon et al., 2012). Without the enforcement of accessibility legislation and the absence of managerial impact on accessibility, it is quite normal for the Web team to ignore the accessibility issue or to pretend they have taken care of it (Pereira et al., 2014). Thus, individual industries and universities should motivate IT practitioners and students by providing them a comprehensive knowledge of accessibility and the difference it makes to disabled users and community commercial organizations (Villena et al., 2014). Moreover, organizations can provide direct incentives to their employees for the accessibility work (Accessibility Sprint Work Group, 2016). Private and public sectors

should stress that accessibility is an important indicator of website quality and without it, the quality of the website will remain low.

Accessibility training for Web developers is still not available in most IT training centers (Lazar et al., 2013; Vázquez, 2014; Gordon & Lujan-Mora, 2015). Thus, organizations should start sponsoring workshops and training courses for their employees and allocate appropriate funding to support accessibility in their reserved budget. On the other hand, stressing the importance of accessibility guidelines should take place at the entry level of Web development courses at universities.

2.5.1.2 Barrier-free design approach

In the U.S., the barrier-free design approach emerged during the civil rights and disability rights movements to protect people with disabilities from discrimination (Keates et al., 2000; Keates et al., 2002). Barrier-free design approaches led to the development of Section 504 of the Rehabilitation Act (US General Services Administration, 1973), Section 508 of the Workforce Investment Act (US General Services Administration, 1998), and the Americans with Disabilities Act or ADA (Kelly, 1995). Moreover, many voluntary guidelines were established to promote the idea of barrier-free design such as WCAG 1.0 and WCAG 2.0.

In addition, a variety of design approaches emerged that facilitated accessibility, including "Rehabilitation Design," "Design by Story-Telling," "Transgenerational Design," "User Pyramid Design Approach," "Inclusive Design", and "Universal Design" (Newell & Gregor, 2000; Keates et al., 2002). These various guidelines (e.g., WCAG

2.0), regulations and design approaches contributed to the development of barrier free accessibility designs for disabled users.

Rehabilitation Design emphasizes the needs of elderly people and people with severe disabilities. However, intended users were dissatisfied and tended to abandon design solutions for several reasons, including functionality, usability, quality, and safety problems (Gardner et al., 1993).

Another design approach, Design by Story-telling, is achieved by understanding the nature of aging, observing older users' activities, exploring design solutions, and evaluating suggested design solutions (Keates et al., 2002). However, this design approach does not take into account users with disabilities.

Transgenerational Design also emphasizes the issue of age, but focuses on developing a product that works for users of all ages from young to old (Keates et al., 2002). However, this design approach also does not consider users with disabilities.

On the other hand, User Pyramid Design is applied by taking into account users' age and impairments. This design approach is based on the assumption that if a product is designed to be accessible by a particular group of users, the resultant product will also be accessible by users with less severe or no impairments. Such a claim implies that a certain group of users will be excluded in developing design solutions.

Inclusive Design was influenced by the claims of User Pyramid Design. Inclusive Design aims to design a product for the largest possible population, but not the entire population (Keates & Clarkson (a), 2003; Keates & Clarkson (b), 2003; Clarkson et al., 2007).

However, it lacks a systematic, ‘human factors’ design protocol. Furthermore, Inclusive Design is more likely to restrict users to those who are merely providing their experiences.

Universal Design is defined by the Center for Universal Design at North Carolina State University as “the design of products and environments to be usable by all people, to the greatest extent possible, without the need for adaptation or specialized design” (The center for universal design, 1997). The Universal Design approach is often employed to make a single product by aiming to satisfy all users, including those with disabilities. However, the ‘one-product-fits-all’ type approach is currently viewed as nearly impossible to achieve (Newell & Gregor, 2000; Nicolle & Abascal, 2001; Abascal, 2002; Kyoko, 2005; Jhangiani & Smith-Jackson, 2007).

Problems surrounding web accessibility guidelines were discussed earlier in section 2.3.1, such as: complex and lengthy nature of the guidelines. Thus, Table 2.4 summarizes and compares the other design approaches as suggested by (Kim, 2010).

TABLE 2.4: Web Accessibility Design Approaches Comparison

Design Approach	Design Aspects	Design Limitations	References
Rehabilitation Design	Developing solutions for people with severe disabilities in their daily life and elderly people	Released products were not qualified enough to satisfy the needs of users Accordingly, a majority of intended users did not purchase the products.	(Gardner et al., 1993; Buhler, 1998; Keates et al.,2002)
Design by Story-telling	Visualization of possible design solutions especially for older users	Lack of awareness in users with disabilities Difficult for designers to comprehend what it is like to be old.	(Moggridge, 1993; Keates et al., 2002)
Transgenerational Design	Product design that enhances the quality of life for users of all ages	Lack of awareness in users with disabilities	(Keates et al., 2002; Pirkk,1994)
User Pyramid Design	Design process by sorting target users based on their capabilities (i.e., severely impaired, reduced strength, or fully capable). Issues related to aging and disabilities are covered.	Design begins by accepting the concept that this design approach cannot satisfy all and should exclude certain groups of users.	(Keates et al.,2002)
Inclusive Design	Product designed for the largest possible population, but not the entire population	It is more likely to restrict users to those who are merely providing their experiences.	(Keates & Clarkson (b) ,2003)
Universal Design	One-product-for-all types design to satisfy all users	It is nearly impossible to achieve from a practical point of view	(Nicolle & Abascal, 2001; Keates et al., 2002; Kyoko, 2005; Jhangiani & Smith-Jackson, 2007)

Source: Kim (2010)

2.5.2 Correction Actions

Before releasing an accessible website it is important to test the code against accessibility guidelines (W3C, 2014). For large companies, beside a development team there is a testing team who check if testing is going hand in hand with the website development in all phases. Once the testers check the website using one or more technique such as automated tools, subjective assessment and testing with real users, the development team has to fix the detected barriers (Gonçalves et al., 2014). However, when it comes to accessibility subject, developers' experiences vary. That is to say, some Web developers have a good technical experience in dealing with accessibility issues while others have only limited experience. Thus, developers with technical savvy do not face obstacles regarding fixing all the detected barriers (Brajnik et al., 2010). On the other hand, novice developers with limited experience who run their own websites will take the responsibility of testing their websites if they choose to produce accessible websites (Vázquez, 2014). Usually, naïve and novice developers choose automated tools to check the accessibility since the subjective assessment and testing with real users are normally beyond their experience and budget (Lazar et al., 2013). However, once these developers want to correct the detected barriers by tools they face another problem regarding guidelines conformance. One common challenge that is usually faced by Web developers is their inability to interpret or understand guidelines to enable accessibility. Within WCAG 2.0 there are many success criteria (SCs) to follow and accessibility barriers to avoid, which would take extensive hours and effort to navigate and understand to insure that their websites are barrier free. Moreover, WCAG 2.0 emphasizes on perfect scores on all SCs in a given priority level to receive the level conformance logo and it ignored the

fact that, in the website development process, developers do not seek perfection; instead, they aim for incremental improvement over versions. Thus, many efforts were made to identify the most common and severe web accessibility barriers based on experts knowledge and experience, pilot studies and surveys that target Web developers. Providing a concise list of most common barriers would simplify checking against guidelines and save time and effort instead of wasting time in studying all type of barriers, since some of them may rarely occur (Hudson, 2010).

The first approach of defining common barriers is based on experts' experience. For example, the Web Accessibility Toolkit (2013) is a project from Cornell University. This project aims to assist college administrators and disability services professionals promote equal access to campus web resources. Web Accessibility Toolkit identifies common web accessibility barriers based on Cornell University's research findings and other experts in the area of accessibility and usability. Another effort was made by the University of Melbourne (2013) which published a list that outlines the 10 most common accessibility problems on the University of Melbourne website. On July 28, 2010 Roger Hudson the founder of Web Usability Company published the article "Ten Common Accessibility Problems" (Hudson, 2010). His company which based in Australia provides services to corporate and government clients regarding usability and accessibility. The author published a list of common barriers that violates WCAG 2.0 based on his experience. However, the author just mentioned the most violated guidelines not the success criteria. Moreover, the previous efforts did not provide any ranking for barriers and only mentioned the common barriers, Table 2.5 summaries the common barriers based on the first approach.

TABLE 2.5: Common Barriers that were Identified Based on Experience

Section 508		WCAG 2.0
Web Accessibility Toolkit (2013)	University of Melbourne (2013)	Roger Hudson (2010)
Graphic images without text descriptions.	Add alt text to images and add labels to forms.	Failure to include text alternatives for images.
Web pages that could not be navigated without a mouse	Use heading tags, table headings and lists.	Use of CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart).
Poor contrast between text and background colors.	Provide sufficient color contrast.	Failure to provide adequate alternatives for other inaccessible content.
Lack of appropriate labels and instructions on data entry forms.	Make each page navigable by keyboard alone.	Failure to use HTML header elements appropriately.
Difficulty navigating when presented with unfamiliar terminology (e.g. Admissions versus registration, course list versus class list, online courses versus online application, etc.).	Allow pausing of animations.	Failure to explicitly associate form inputs with their labels (or use the input title attribute).
	Add skip to content links.	Failure to ensure sufficient difference between foreground (text) color and background color.
	Add a meaningful page title.	Failure to identify data tables with summary attribute or caption element.
	Avoid "click here" link text.	Failure to mark-up data tables correctly.
	Keep navigation consistent.	Failure to ensure sites can be used without the mouse.
	Avoid pop-up windows.	Use of "onchange" event handlers with select menus.

The second approach is based on the outcomes of pilot studies and experiments to define the most common barriers. The Society of IT Management (Socitm, 2008) published the report "A world denied: a supplement for Better connected 2008 on website accessibility" on May 2008. This report provided the results of a set of accessibility testing carried out and included a list of the five most common accessibility errors regarding WCAG 1.0.

According to this report, these five errors account for 76% of all website accessibility failures. Another project was conducted in June 2007, by Cornell University (Erickson et al., 2007) to evaluate a sample of community college websites for accessibility and usability by students with and without disabilities. This research based on Section 508 checkpoints and revealed some of the most common barriers to access websites. The common barriers identified by Socitm and Cornell University are illustrated in Table 2.6.

TABLE 2.6: Common Barriers that were Identified Based on Pilot Studies run by Socitm and Cornell University

WCAG 1.0 Socitm (2008)	Section 508 Cornell University (Erickson et al., 2007)
No alternative text for images.	Timed response requirements cannot be modified by user.
Inappropriate use of JavaScript.	Accessible alternative to proprietary format not provided.
Errors in simple data tables.	Data tables do not have marked headers.
Errors in complex data tables.	Frames do not have titles.
Use of features with a lack of accessible alternatives.	Skip navigation links not provided.
	Form element not labeled and not keyboard accessible.
	Scripted elements are not accessible.
	Alt text for images missing or inappropriate.

In 2005, the Pennsylvania State University published the webpage "Most Common Web Standards and Accessibility Errors" that violates Section 508 (Pennsylvania State University, 2005). Table 4.7 illustrates the common barriers along with their frequencies based on their pilot study ranked in a descending order from most severe one to least severe.

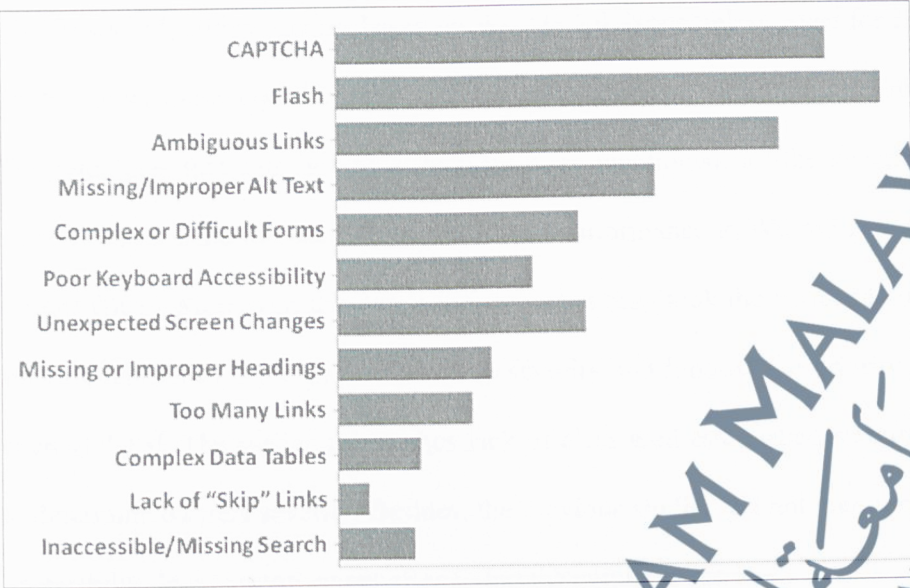
TABLE 2.7: Common Barriers Along with their Frequency Based on Pilot Study run by Pennsylvania University

Barrier Type	Frequency
Image should have valid ALT	60% (most severe)
Skip repetitive links	30%
Form control should have a label	8%
Color should not be essential	8%
Multimedia should have equivalent audio descriptions	1%
No javascript links should be used	0.1% (least severe)

Source: (Pennsylvania State University, 2005)

The third approach basically based on surveys that target Web developers to determine the most common barriers they face. WebAIM is an organization within the Center for Persons with Disabilities (CPD) at Utah State University. Since May 2012, WebAIM has conducted four online surveys of preferences of screen reader users. On May 31, 2012 the results of the fourth survey were published online (WebAIM, 2012). In this survey there were 1782 valid responses from Web developers. The respondents were asked to select their most, second most, and third most problematic barriers from a list. Each selected barrier was given a weight, and the results are shown in Figure 2.2 and ranked in Table 2.8. According to this survey, 21.9% of respondents listed inaccessible Adobe Flash content as the most difficult barrier encountered, with CAPTCHA closely behind (21.6%). When compared to responses to these questions over the previous three surveys, the order and indicated difficulty for the items in this list are largely unchanged. However, no information was provided on how they assigned weights to the barriers.

FIGURE 2.2: Most Severe Barriers in WebAIM’s Screen Reader: Fourth User Survey



Source: (Pennsylvania State University, 2005)

TABLE 2.8: Barriers Severity Ranking in WebAIM

Barriers ranking order	Barrier Type
1 (most severe)	Inaccessible Adobe (Macromedia) Flash content.
2	CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart).
3	Links or buttons that do not make sense.
4	Images with missing or improper descriptions (alt text).
5	Screens or parts of screens that change unexpectedly.
6	Complex or difficult forms.
7	Lack of keyboard accessibility.
8	Missing or improper headings (h1, h2, etc.).
9	Too many links or navigation items.
10	Complex data tables.
11	Inaccessible or missing search functionality.
12 (least severe)	Lack of "skip to main content" or "skip navigation" links.

Although a great effort has been made in identifying common barriers based on the previous approaches, the results vary from an experiment to another depends on the context and the targeted websites. Furthermore, most of the Web based approaches are

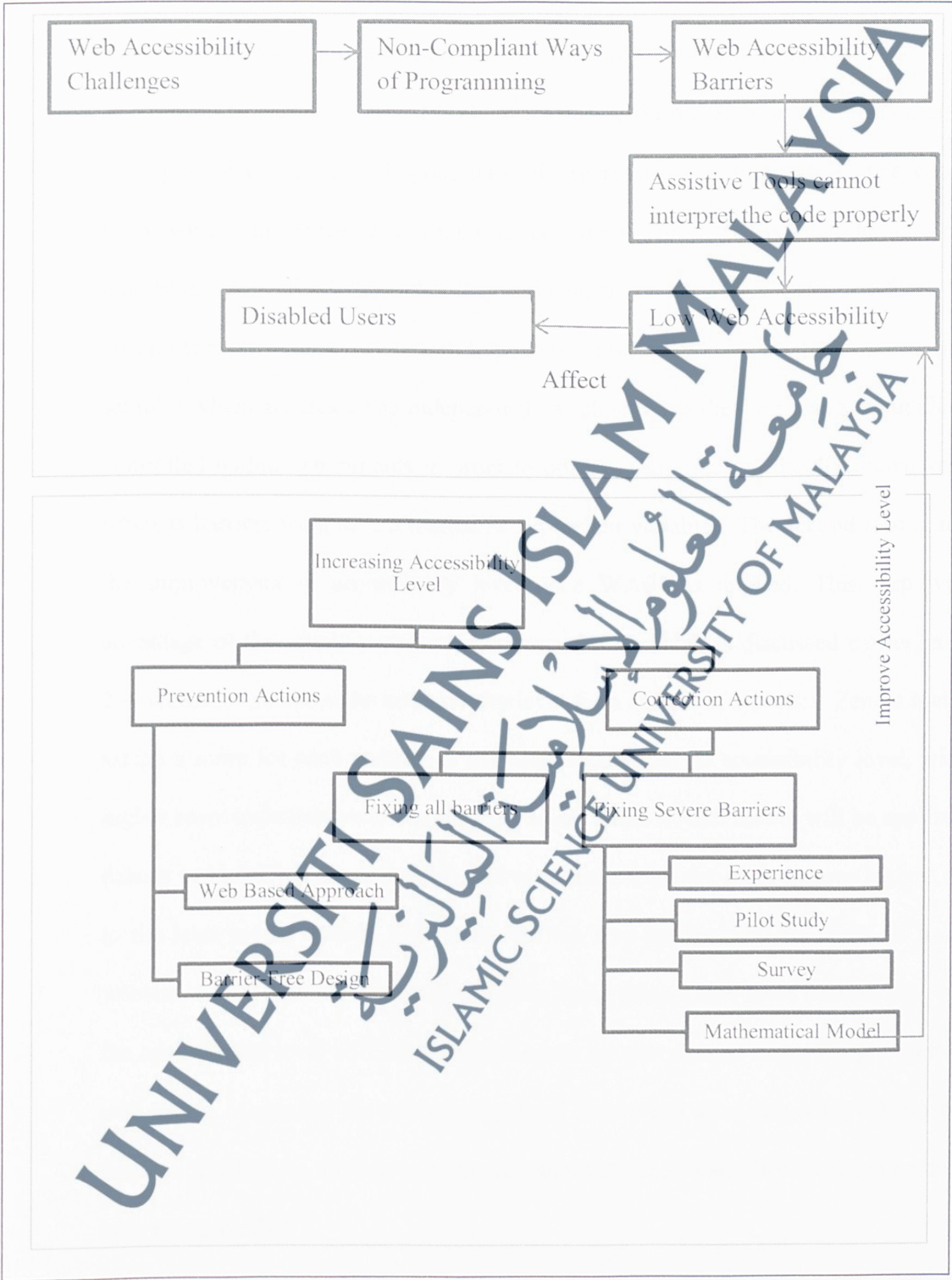
based on Section 508 except for Socitm (2008) which based on WCAG 1.0 and Hudson (2010) who identifies barriers based on WCAG 2.0. Nonetheless, even for Hudson (2010) study that was based on WCAG 2.0, the barriers ranking were ignored. Additionally, only the guidelines that can face most barriers were mentioned not the success criteria that were violated since the basis for determining conformance to WCAG 2.0 are the success criteria (SCs). Moreover, the previous approaches only took the frequency of barriers in a specific dataset to determine the barriers severity and ignored the priority level that has been violated. The earlier approaches lack of clear well established mathematical model to determine barriers severity. Besides, the previous studies did not measure the degree of accessibility level improvement after fixing the common barriers.

2.6 THEORETICAL FRAMEWORK

This work proposes a mathematical based metric that reveals the most prevalent and severe web accessibility barriers that affect websites in a certain sample, and then rank them based on their severities. Once the barriers are ranked, the accessibility level of a set of websites will be measured after removing or fixing the severe barriers in order to measure the improvement in the accessibility level for disabled users, specifically blinds deaf and elderly users. The theoretical framework of this research encompasses all the related concepts to improve the accessibility level. The theoretical framework is diagrammatically represented in Figure 2.3. The web accessibility challenges that were discussed in Section 2.3, can lead to non-compliant ways of coding that violates WCAG 2.0. According to (W3C, 2016) this causes accessibility barriers in the design that prevent assistive tools from interpreting the code properly which results in low accessibility level

that directly affect disabled users' interaction and accessing the Web. There are prevention actions to decrease the possibility of barriers while designing a website as well as correction actions to fix the barriers once occurred. The prevention actions are mostly followed by large firms that endorse their own legal standards or follow accessibility legislations and guidelines. These firms and companies have experienced Web teams who are familiar with the accessibility issues unlike regular developers who run their own websites. On the other side, the correction actions aim at fixing the barriers once they are detected at the testing phase. Large companies compel their Web teams to fix all the detected errors to gain WCAG 2.0 level conformance logo. However, for small Web businesses and novice developers who have limited budget and experience in designing accessible websites, they do not follow prevention actions and they struggle in producing accessible free design. They do not know where to start fixing and what needs to be fixed first. Thus, many efforts were made to determine the most common barriers that threaten the accessibility so the Web developers can fix them to increase the accessibility of websites. These efforts took into account that developers aim for a pragmatic improvement over versions. Determining severe and common barriers can help Web developers to start fixing these severe barriers and to avoid them in the future (Hudson, 2010). However, the majority of the correction approaches lack of a clear mathematical model. So, this work took a step in proposing a mathematical based metric that ranks accessibility barriers based on their severities and influence on the accessibility issue.

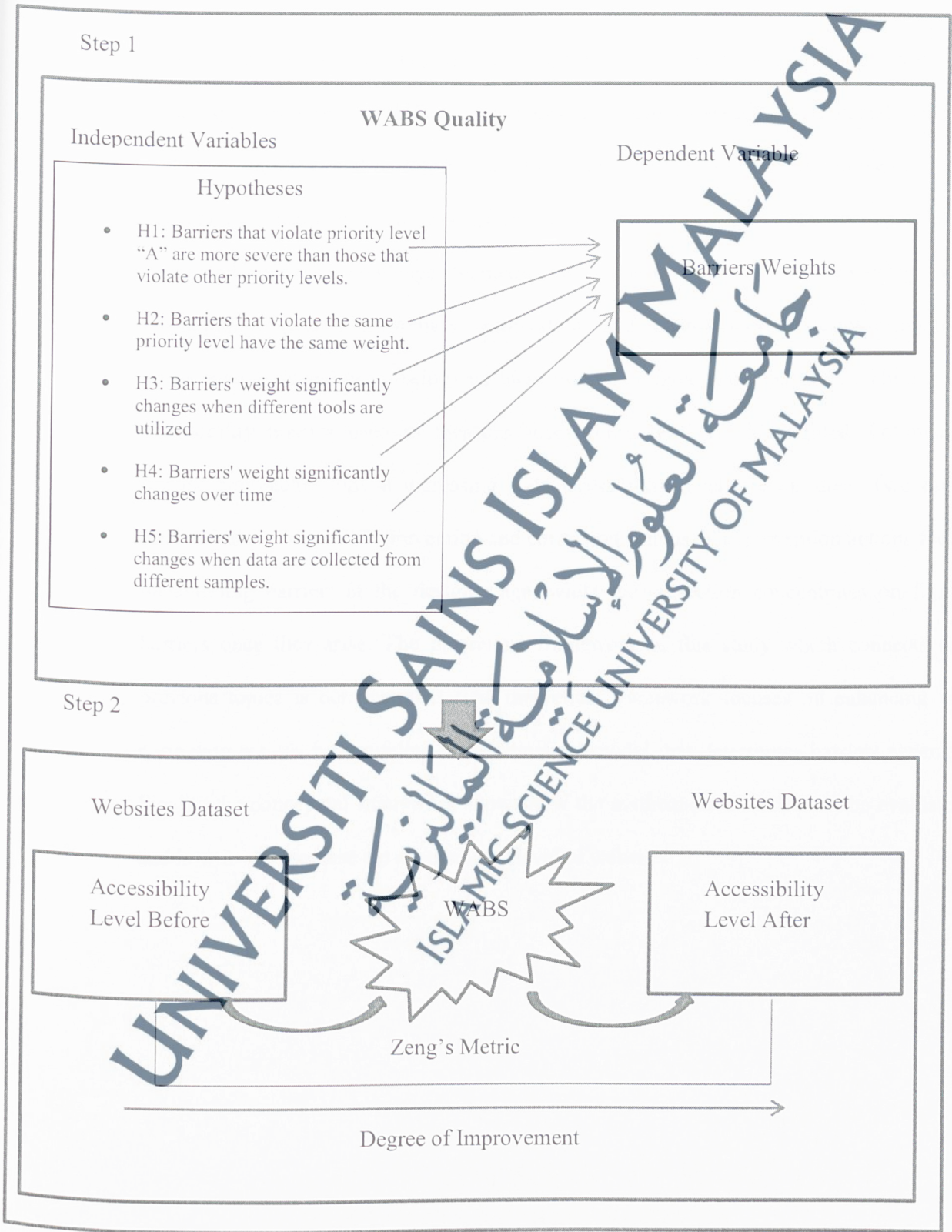
FIGURE 2.3: Research Theoretical Framework



2.7 CONCEPTUAL FRAMEWORK

This work proposes a quantitative metric that ranks accessibility barriers based on their severities. The metric is named Web Accessibility Barrier Severity (WABS) metric. The conceptual framework is diagrammatically represented in Figure 2.4. The conceptual framework of this research encompasses two steps. The first step is validating the metric quality in terms of validity, reliability and sensitivity. Five hypotheses are formulated to check the metric quality. Research hypotheses covers the automated tools, time span and samples which are treated as independent variables since they are being manipulated and controlled within experiments in order to observe the effect on the dependent variables, whereas barriers weights are treated as dependent variables. The second step is measure the improvement in accessibility level once WABS is applied. This step will take advantage of the metric proposed by (Zeng, 2004) which is discussed earlier in section 2.4 in order to measure the accessibility level for a dataset of websites. Zeng's metric will assign a score for each website in the dataset based on its accessibility level, where the higher score indicates lower accessibility level. Then WABS metric will be applied to the dataset to identify barriers weights and rank them from the most severe (highest weight) to the least severe barrier. Following, the top five barriers will be removed from each website in the evaluated dataset to simulate fixing them. Once these barriers are removed, the accessibility level will be measured again for that dataset using Zeng's metric. This will enable comparing the accessibility level before and after removing the most severe barriers. In addition, the degree of improvement in the accessibility level can be measured after using WABS.

FIGURE 2.4: The Conceptual Framework of This Research



2.8 CHAPTER SUMMARY

This chapter reviews literature and researches relevant to the work. First, an overview of disability types is presented. Four main types of disabilities are identified, namely visual, auditory, cognitive, and motor. Second, the chapter reviews the negative impact of web accessibility challenges on accessibility level and discusses the primary problems associated with each challenge. Standards and guidelines, during website's design and development (user's awareness, motivation and training) and during evaluation (automated tools and user testing) are identified as the three main challenges. Third, web accessibility metrics used to measure accessibility level are highlighted. Following, diverse approaches aim at increasing web accessibility level are outlined. Two main approaches are identified: prevention and correction actions. The prevention actions focus on avoiding barriers at the design stage, while the correction concentrates on fixing barriers once they arise. The theoretical framework of this study which connects the previous topics is demonstrated. The theoretical framework focuses on enhancing the correction actions by providing a mathematical model that determines barriers severity. Finally, the conceptual framework shows how the mathematical model will be evaluated and how it will increase the accessibility level of websites.