

DCWA: Testing tool for Detection and Correction in Web Accessibility

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

Master of Technology
(INFORMATION SYSTEMS)

SUBMITTED BY
Aman Jain
(01/IS/10)

Under the Guidance of:

N.S. Raghava
ASSOCIATE PROFESSOR
Department of Information Technology



Department of Information Technology
Delhi Technological University, Delhi - 110042
2010-2012

CERTIFICATE

This is to certify that **Mr. Aman Jain (01/IS/10)** has carried out the major project titled **DCWA:Testing tool for Detection and Correction in Web Accessibility** as a partial requirement for the award of Master of Technology degree in Information Systems by Delhi Technological University. The major project is a bonafide piece of work carried out and completed under my supervision during the academic session 2010-2012. The matter contained in this report has not been submitted elsewhere for the award of any other degree.

Project Guide

N.S. Raghava

Associate Professor

Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

ACKNOWLEDGEMENTS

I express my gratitude to my guide N S Raghava, Department of Information Technology, Delhi Technological University, for the valuable support and guidance he provided in making this thesis successful. It is my pleasure to record my sincere thanks to my respected guide for his constructive criticism and insight without which the project would not have shaped as it has. I would like to thank Prof. O P Verma, Head of the Department of Information Technology, Delhi Technological University for his ultimate support. Your insights, advice and constant encouragement have been critical. I must thank Mr. Nikhil Maheshwari, Assistant system Engineer,IBM, for his support while working in this domain. I would also like to thanks my all family members especially my father, who have been a source of encouragement and inspiration throughout the duration of the project. I would also like to thank the entire DTU family for making my stay at DTU a memorable one.

Aman Jain
(01/IS/10)

ABSTRACT

Web is most powerful source of information but Web users can even have disability, which may affect them permanently or temporarily. Some have more difficulties than others or simply if we say that person with disability (PWD) have their own problem accessing web. And there is a question comes in usability and accessibility of web for PWD.

Nowadays Web-designers have focused on designing the interface that is eye pleasant due to which the issues of web accessibility have been poorly addressing and it becomes problem because users do not have the equivalent ability to accesses and exploit the web content. Therefore several guidelines and recommendations have been introduced so that web content can be made more accessible and usable to all users

Lots of initiatives have been taken in this direction of making websites accessible and usable but still the problem of the accessibility of Web pages is unattended by many sites.

Evaluation of web sites becomes compulsory to determine that weather websites meets accessibility guidelines or not and this is done through web accessibility evaluation tools that are software programs or online services that help to determine this.

Web accessibility evaluation tools are significantly able to reduce the time and effort needed in carrying out evaluations. Moreover if we use these tools carefully throughout whole designing, implementation and maintenance part of development then these tools can help their users in preventing accessibility barriers, reporting the encountered barriers means they can improve overall quality of websites.

Web Accessibility Evaluation tools can be classified as automatic and semi automatic. The automatic tools allow developers and designers to quickly evaluate websites for accessibility Lots of Evaluation tools are available in market but they have some drawbacks like they detect the accessibility issues and generate tips for developers but they don't have any efficient correction mechanism in built with them.

A new tool DCWA is introduced that will do dual task of detecting the web accessibility issues as well as correcting them too that reduces the burden of developer to manually checking tags one by one. This multi function in DCWA makes it better than

other available tools. Detection and corrections both are available in previous resources separately but the combination of these two is a key success to this tool.

There are some matrixes on which the performance of various web accessibility evaluation tools is measured so there is a comparison between various evaluation tools. In general, the objectives of this research are:

- To develop a testing tool **DCWA** for detection and correction in web accessibility.
- To provide a comparative study on various web accessibility evaluation tools.

Contents

CERTIFICATE	i
List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Access The World	1
1.1.1 Little About Web Environment	2
1.1.2 Website	2
1.1.3 Web Browser	3
1.1.4 World Wide Web Consortium(W3C)	3
1.2 Disabled Users	4
1.2.1 Disability	4
1.2.2 Laws & Rights	5
1.3 Meaning To Accessibility	6
1.3.1 Definition Of Terms	7
1.4 Some Steps towards Web Accessibility	8
1.4.1 Measure Involvement	8
1.4.2 Some Guidelines	8
1.5 Web Accessibility Evaluation	10
1.6 Thesis Organization	11
2 Literature Review	12
2.1 Related Work and Background	12
3 Web Accessibility Guidelines And Evaluation Tools	16
3.1 Guidelines to Follow	17
3.2 An Accessibility Guide	18
3.2.1 WCAG 1.0	18
3.2.2 WCAG 2.0	20
3.3 Web Accessibility Evaluation Tool	21
3.3.1 Introduction	23
3.3.2 Types of Evaluation Tools	23
3.3.3 Dos by Evaluation Tool	24
3.3.4 Don'ts by Evaluation Tool	24
3.3.5 Usage of Evaluation Tool	24
3.3.6 Features of Evaluation Tools	25

3.3.7	Famous Known Tools	26
3.3.8	New About DCWA	28
4	Detection & Correction In Web Accessibility	29
4.1	Motivation	29
4.2	Description of System	30
4.3	Parsing	31
4.3.1	Parsing web page	31
4.3.2	Results	32
4.4	Detection	32
4.4.1	Detecting web Accessibility Issues	33
4.4.2	Result	33
4.5	Correction	34
4.5.1	Correction Based on Detected Issues	34
4.5.2	Result	35
4.6	Process Workflow of Tool	36
5	A Delivery Model for Accessibility Evaluation	37
5.1	Demo Of Tool	37
5.2	Proposed Approach	42
6	Conclusions And Open Research Lines	47
	References	51

List of Figures

1.1	Screen shot of web browser with its components	3
1.2	Logo of W3C	4
1.3	Internationally recognized symbol for accessibility	7
4.1	parsing source file	32
4.2	Detection	33
4.3	Correction of WCAG violation detected page	35
4.4	Process workflow of DCWA	36
5.1	click here to run tool	37
5.2	Loading of tool	38
5.3	Browse your Page	38
5.4	Input Page	39
5.5	popup window showing generation of detection of file	40
5.6	Violation Detected Page	40
5.7	popup window showing that victim elements have been corrected	41
5.8	Violation Free Page	41

List of Tables

1.1	categories of Definition of “Disabilities”	6
3.1	Guidelines from WCAG 2.0	22

Chapter 1

Introduction

Berners-Lee invented the World Wide Web that has brought a change in the world in consuming and producing information for various purposes. A new term was coined Website that has been created to fulfill various needs and activities for different kinds of users e.g. information gathering, transactions, communication, entertainment and many more.

Now imagine the world without web or the life of those people who are disabled or partially disabled. They are still underprivileged and need more caring.

This research is an effort to check out the accessibility issues and providing fixing mechanism for them. It also focuses on some guidelines to follow up.

1.1 Access The World

In accordance to Tim Berners-Lee, he had a big idea in mind when he and Robert Cailliau invented the Web: a “common information space in which we communicate by sharing information.”[1]

In 1989, while both were working at CERN (the European Organization for Nuclear Research), they made proposals for hypertext systems. In 1990 Lee and Robert joined forces and wrote a joint proposal in which the term “World Wide Web” is used for the first time. And then in late 1990 and early 1991, Tim Berners-Lee could write the first web browser.

Now the world is a single entity and in this era of information age it is just one click

away to get most of the information we are desired for. People are getting connected to each other with this blowing information over a common path known as World Wide Web provides a user interface to the internet, which is made up of number of computers connected to each other with a network.

1.1.1 Little About Web Environment

The core principal of Web is based on the navigation between documents (Web Pages) by clicking hyperlinks using a program called browser. A Web page is written in simple text file by using tags of markup language (HTML) to decide the layout and formatting of page. There are some acronyms used when we talk about Web.

- **WWW:** World Wide Web also known as W3, or just say “The Web”.
- **HTML:** HyperText Markup Language [2]. Used for displaying the contents on web.
- **URL:** Uniform Resource Locator [3]. It is a standard for specifying an object on the Internet, such as a file or image. It’s the thing in the “Address bar” on your web browser. e.g. <http://www.w3.org/>
- **HTTP/HTTPS:** HyperText Transfer Protocol/Secure Server HTTP.
- **Hyperlink:** To navigate from one page to other.
- **Surfing:** Accessing the web.
- **Webpage:** A document written in HTML.

1.1.2 Website

A website is a collection of related web pages containing content such as text, images, video, audio etc. To run a website we need at least one web server that is a place where web is stored and that must be accessible via a network such as the Internet or a private local area network through an Internet address known as a URL ,Uniform Resource



Figure 1.1: Screen shot of web browser with its components

Locator. All accessible websites collectively constitute the World Wide Web. URL looks something like this <http://www.w3.org/WAI/gettingstarted/Overview>.

Let's go through this URL:

- **http://**: It is the default protocol for browsing the web.
- **www.w3.org**: It is the web address.
- **WAI/gettingstarted/Overview**: It is where the document is stored server.

1.1.3 Web Browser

Software used for running website by providing user friendly graphical interface (GUI). There are many Web Browsers available in market and some of them are Firefox, Netscape, Opera, Internet Explorer, Safari etc.

In Figure 1.1, we are showing the basic elements of browser and these are present in almost all browsers available in market.

1.1.4 World Wide Web Consortium(W3C)

The World Wide Web Consortium is an international community where a full-time staff, member organizations, and the public work together to develop Web standards that are



Figure 1.2: Logo of W3C

followed worldwide. It was Lead by Web inventor Tim Berners-Lee and CEO Jeffrey Jaffe, The main aim of W3C's to lead the Web to its full potential developing protocols and guidelines that ensure the long-term growth of the Web.

W3C works on following principles:

- **Web for All:** Web has some social values that it enables human communication, commerce, and opportunities to share knowledge. W3C's primary goals is to make these benefits available to all people, whatever their hardware, software, network infrastructure, native language, culture, geographical location, or physical or mental ability.
- **Web on Everything:** There are immense numbers of devices that can access the web and this number is rapidly growing day by day. Some of them are Mobile phones, smart phones, personal digital assistants, interactive television systems, voice response systems, kiosks and even certain domestic appliances can all access the Web.

1.2 Disabled Users

Disabled users are those people who have one or more than one type of disability to access and exploit web content. They are commonly known as person with disability (PWD).

1.2.1 Disability

“Disability” [4] implies that a person is not able to do some things or anything (Baldwin, 2000) [5]. Most of the definitions of disability focus on only one aspect of disability

and not upon the criteria which made a person disable. The basic criterion for successful interaction between user and system is the matching of their capabilities. If this matching is not there then user would not be able to access the system. Four main categories of disabilities that effect person's ability to use the web are: [6]

- **Mobility:** This includes inability to move, insufficient dexterity to operate a mouse or a keyboard, inability to control unwanted movement, and lack of limbs.
- **Hearing:** This includes inexact hearing to diminished hearing to no hearing at all.
- **Vision:** This includes partial or total blindness and colorblindness. This is the most common disability people are suffering from.
- **Cognition and Learning:** This includes various difficulties reading, understanding, staying focused, remembering, and writing.

Most of the studies say that a significant number (20%) of our population has disability of some kind. These people are not capable of using internet unless any additional support is provided to them. Businesses are unknowingly excluding this 20% population which can be their potential customers. Similar is the case with University, Schools and Government entities who are unaware that their websites are not accessible and they are breaking laws. Many people consider that disability leads to failure but it is inappropriate since they neglect the abilities the person may possess. Each of these disabilities requires different adaptation in order to make web content accessible to these. These adaptations are beneficial to disabled and non-disabled both.

1.2.2 Laws & Rights

Different countries have some laws to enhance Accessibility in different environments. Few important laws related to Accessibility [7][8] are explained in this section:

- **India:** The Persons With Disabilities (Equal Opportunities, Protection of Rights and Full Participation) Act, 1995, commonly known as The PWD Act.

Table 1.1: categories of Definition of “Disabilities” [9]

Category 1: Individual as a unit of analysis		Category 2: Society as a unit of analysis	
Biological or medical model	Function or Rehabilitation Model	Environmental Model	Human Rights Model
Emphasis on attributes in the individual	Emphasis on promoting or restoring fuller functioning in the individual.	Attention directed to ecological barriers: social, institutional and legal, which can result in disabilities	Focus is on the rights to which all people, including people with disability, are entitled

- **United States:** The Americans with Disabilities Act (ADA) 1990 Section 508 of the Rehabilitation Act 1973 Individuals with Disabilities Education Act (IDEA).
- **United Kingdom:** The Disability Discrimination Act (DDA) 1995 Special Educational Needs and Disability Act (SENDA) 2001.
- **Australia:** The Disability Discrimination Act (DDA) 1992.
- **Japan:** Human Resources Development Promotion Law.
- **China:** Law of the People’s Republic of China on the Protection of Disabled Person. The definitions of disability are mainly given based on two factors, individual and society which are further divided in four models (Rioux, 1997) [9].

1.3 Meaning To Accessibility

Accessibility is the “ability to access” and benefit from some system or entity. Accessibility is often used to focus on persons with disabilities or special needs and their right of access to entities, often through use of assistive technology. The word Accessibility should not be confused with usability which is used to describe the extent to which a product (e.g., device, service, and environment) can be used by specified users



Figure 1.3: Internationally recognized symbol for accessibility

to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. Accessibility is firmly related to universal design when the approach involves “direct access”. This is about making things accessible to all people.

1.3.1 Definition Of Terms

- **Accessibility:** Accessibility means making access to content possible. For example Mr. X is unable to access the statistics and other information contained within the image that was not assigned alternative text. So we can say that content is inaccessible to Mr. X. [10]
- **Usability:** It means creating content in a way that helps users be more effective and efficient. Content can be possible for someone to access, but it might require inefficient or confusing interactions. [10]
- **Availability:** Improving availability means to make available the access to technology wherever users happen to be. The technology that users need to access the web is often not available to them. Some time many people do not own their own computers and rely on public computers, such as those at libraries and schools, for access. It happens because access technology is not installed on most computers for many reasons. [10]
- **Web Accessibility:** Web accessibility [11] means everyone can use the web including the people with disabilities. The purpose of Web accessibility is to provide environment so that everyone can perceive, understand, navigate, interact

and contribute to the Web. Web Accessibility has now become a major area for research.

1.4 Some Steps towards Web Accessibility

Providing Accessibility to web has become area of interest and also duty. So some initiative has been taken in this direction.

1.4.1 Measure Involvement

W3C has taken initiative in the direction of Web Accessibility, known as Web Accessibility Initiative (WAI) [12], to develop standards and guidelines for web browsers, authoring tools, web content etc. Guidelines for Web Accessibility are known as Web Content Accessibility Guidelines (WCAG) [13].

1.4.2 Some Guidelines

There Are some guidelines that should be followed to make website accessible.

WCAG 1.0: Web Content Accessibility Guidelines 1.0 (WCAG 1.0) was published in May 1999. It was the major step taken in the direction of improving web accessibility. It was divided in 14 guidelines and numerous checkpoints, which were used to check the accessibility of a web page. There were 3 priorities or level of conformance to check whether the web page is following the guidelines or not.

- **Priority 1 or Level a conformance:** basic requirement
- **Priority 2 or Level AA conformance :** better accessibility and removal of significant barriers
- **Priority 3 or Level AAA conformance:** improvements to web content accessibility

WCAG 1.0 was very HTML specific in nature. Due to the limitations of WCAG 2.0, the development of WCAG 2.0 was considered.

WCAG 2.0: WCAG 2.0 was published on December 11, 2008 and it is applicable to advanced technologies. It is easily available, usable, understandable and testable with automated testing tools as well as human evaluation. WCAG 2.0 is a four layered structure. Now we will discuss these layers one by one.

- The **first layer** consists of four design principles collectively known as POUR: Perceivable, Operable, Understandable, and Robust.
- The **second layer** is a set of twelve guidelines with basic goals of accessibility.
- The **third layer** is a set of success criteria similar to user requirements.
- The **fourth layer** consists of sufficient and advisory techniques required to meet the success criteria. These techniques can be developed inside as well as outside W3C.

Principles: The foundation for Web accessibility is based on four principles: Perceivable, operable, understandable, and robust known as POUR

- **Perceivable:** Web content and inefficient formation should be available to senses either through the browser or assistive technologies.
- **Operable:** Web content like controls and elements should be operable with keyboard, mouse, or an assistive device.
- **Understandable:** No ambiguity or confusion should exist in web content.
- **Robust:** Web content should be accessible through various available technologies.

Guidelines: The author should follow twelve guidelines (Table 2: Guidelines from WCAG 2.0) in order to make web content more accessible to users. These guidelines are not testable directly but provide foundation for better understanding of success criteria and implementation.

Success Criteria: Guidelines stated in WCAG 2.0 are testable through the success criteria provided. Success criteria should be performed during accessibility testing along with requirement and conformance testing. Based on the need of different

organization, levels of conformance are defined in three parts: A (lowest), AA (medium), and AAA (highest).

Sufficient and Advisory Techniques: Each guideline and success criteria can be achieved by some techniques documented by working group. The techniques fall into two categories: sufficient and advisory.

All the four layers work together to provide guidance on how to make web content more accessible. The WCAG describes technique for making Web content accessible to everyone including disabled people. “Web content” generally means information contained in a Web page and or application, including text, images, graphics, sounds etc.

1.5 Web Accessibility Evaluation

Web Accessibility Evaluation is a very hot topics for researchers in this direction because the use of web is increasing day by day so the users of web contents are also increasing in same manner but developers and engineers are more conscious about designing part rather than making it fully accessible. For making web page accessible it must follow some guidelines. Web Accessibility evaluation tools are software or web application who checks warpage against these guidelines. we will discuss about this in detail in next chapters.

1.6 Thesis Organization

Our whole thesis can be organized in to following chapters:

- An Overview of prior work done in the field of Web Accessibility, related to our work, is described in Chapter 2.
- In our Chapter 3, we understood various web accessibility guidelines (WCAG) and also got to know about the web accessibility evaluation tool that is actual motivation for our work.
- In our Chapter 3, we understood various web accessibility guidelines (WCAG) and also got to know about the web accessibility evaluation tool that is actual motivation for our work.
- A tool that provides both detection as well as correction in a web page (DCWA) is described in Chapter 4.
- Our final Delivery model of our system is described in Chapter 5.
- Conclusion and some open research lines for the future from our research work are described in Chapter 6.

Chapter 2

Literature Review

Literature reviewing is very important part of any research work because it provides existing research in corresponding domain. Here I present some existing research on Web Accessibility and web accessibility evaluation procedure.

2.1 Related Work and Background

In 2006, Davis Sloan of University of Dundee presented “**The Effectiveness of the Web Accessibility Audit as a Motivational and Educational Tool in Inclusive Web Design**” [12]. The development and evaluation of a Web accessibility auditing methodology with the dual aims of accurately identifying accessibility barriers present in a Web site, and presenting the audit findings and recommended actions in a way that informs, educates and engenders an improved understanding of accessibility amongst the audience, was performed.

In 2006, Omar Kheir of University of York presented “**The Accessibility and Usability of Websites: Relationships between Measures from Users, Experts and Guidelines**” [13]. In this research, author tried to aware the web developers to follow accessibility guidelines by the accessibility study conducted on various web environments. While going through this thesis, we got an idea to make the accessibility procedure easy so that developers can follow it.

In 2006, Matt Huenerfauth of University of Pennsylvania presented “**Generating**

American Sign Language Classifier Predicates For English-To-ASL Machine Translation” [14]. An English-to-ASL MT design capable of producing classifier predicates was created in this project. The classifier predicate generator inside this design had a planning-based architecture that uses a 3D “visualization” model of the arrangement of objects in a scene discussed by the English input text. This generator would be one pathway in a multi-path English-to-ASL MT design; a separate processing pathway would be used to generate classifier predicates, to generate other ASL sentences, and to generate animations of Signed English (if the system lacked lexical resources for some input). During the study of this project we thought of developing a system accessible to everyone including the persons with disability irrespective of their language.

In 2007, Scott Hollier of Curtin University of Technology presented “**The Disability Divide: A Study into the Impact of Computing and Internet-related Technologies on People who are Blind or Vision Impaired**” [15]. The purpose of this study was to find the reasons behind this digital divide for people with disabilities and provide solutions. The investigation into this ‘disability divide’ initially examined the historical significance of the social construction of disability, the developments of computing and Internet-related technologies and the evolution of associated government and corporate policies.

In 2009, Julie A. Smith of Capella University presented “**Developing Web Accessibility: Section 508 Compliance of Post-Secondary Educational Web Site Home Pages**” [16]. Web accessibility of education department home pages of institutions accredited by the National Council for Accreditation of Teacher Education (NCATE) was studied. A multiple-methodological approach based on the literature and U.S. Section 508 and Web Content Accessibility Guidelines (WCAG) was used to further understand accessibility issues relating to higher education Web pages and visually-impaired Web users using screen readers.

In 2010, Lourdes Moreno of University of Madrid presented “**AWA, Methodological Framework in the Accessibility Domain for Web Application Development**” [17]. The methodological support AWA (Accessibility for Web Applications) is presented in this thesis. AWA provides a workspace in order to include the accessibil-

ity requirement in the organizations devoted to web application development. AWA provides guidance to engineering professionals to incorporate accessibility requirements from different perspectives: (a) in organizations and businesses, integrating accessibility and quality policies, (b) in the development process following a methodological approach that provides systemization in the integration of accessibility from the outset and, finally, (c) following a User-Centered Design (UCD) approach that places the user as a the main figure and makes him/she participate in the design process.

In 2011, Eduardo Hideki Tanaka, IC/NIED/UNICAMP, Brasil, Helosa Vieira da Rocha IC/UNICAMP, Brasil has presented “**Evaluation of Web Accessibility Tools**” [18] in Brazilian Symposium on Human Factors in Computer Systems & 5th Latin America Conference on Human Computer Interaction. This paper presents some findings of user tests done in order to evaluate four web Accessibility tools. To provide a little more information about the pros and cons of accessibility tools, this paper presents some findings of user tests done in order to evaluate four different Web accessibility tools.

In 2012, Thi-Huong-Giang of VuHanoi University, Vietnam of Science and Technology, Dat Trinh Tuan Hanoi University, Vietnam of Science and Technology, Van-Hung Phan Micronet Group 2 Villa E, Vietnam has presented “**Checking and correcting the source code of web pages for accessibility**” [19], An International Conference paper in IEEE. This paper aims to propose and experiment a framework for checking and correcting websites for accessibility.

In 2011, Ndia Fernandes, Rui Lopes, Lus Carrio, LaSIGE/University of Lisbon, Portugal presented “**On Web Accessibility Evaluation Enviroments**”, [20] in W4A 2011 - Technical Paper, March 28-29, 2011, Hyderabad, India. This paper details an experimental study designed to understand the differences posed by accessibility evaluation in the Web browser.

In 2010, Christopher Bailey & Dr. Elaine Pearson, Teesside University Accessibility Research Centre, School of Computing, Middlesbrough, U.K. has presented “**An Educational Tool to Support the Accessibility Evaluation Process**” [21] in W4A2010 - Communication, April 2010, Raleigh, USA co-Located with the 19th In-

ternational World Wide Web Conference. This paper describes the development of a web accessibility Knowledge management tool, known as the Accessibility Evaluation Assistant (AEA), designed to assist novice auditors in the process of an accessibility evaluation. The tool is designed primarily for undergraduate and postgraduate computing students with limited knowledge of web accessibility, limited opportunity to study accessibility and limited access to expert reviewers or disabled user groups.

In 2011 , Theofanis Oikonomou , Nikolaos Kaklanis , Konstantinos Votis, Grammati-Eirini Kastori , Nikolaos Partarakis and Dimitrios Tzovaras, Informatics and Telematics Institute Centre for Research and Technology Hellas, Thessaloniki, Greece Foundation for Research and Technology-Hellas, Institute of Computer Science, Heraklion, Greece has presented “**WaaT: Personalised Web Accessibility Evaluation Tool**” [22] in W4A2011- Microsoft Challenges. In this paper they introduced an advanced and new personalized Web accessibility evaluation methodology, allowing for Web content accessibility evaluation regarding different selectable disability profiles (impairments, personas) as well as Assistive technologies and devices. We define an evaluation approach based on the Web Content Accessibility Guidelines 2.0 and the Barrier Walk through, with the goal of providing support to Web developers and designers to conduct rapid, yet specialized, accessibility assessments focused on different disability types and user preferences for Web applications.

In 2011, Dr. Elaine Pearson, Christopher Bailey & Dr. Steve Green Teesside University Accessibility Research Centre, School of Computing, Middlesbrough, UK has presented “**A Tool to Support the Web Accessibility Evaluation Process for Novices**” [23] in ITiCSE’11. The Accessibility Evaluation Assistant (AEA) is designed to assist novice auditors in the process of an accessibility evaluation of websites. This paper focuses on the evaluation of an educational web accessibility knowledge management tool which aims to support novice evaluators. It is developed specifically for undergraduate computing students and incorporates a structured walkthrough method to guide the novice auditor through the process of an accessibility evaluation.

Chapter 3

Web Accessibility Guidelines And Evaluation Tools

To provide Web accessibility is a way to ensure that a large range of users can access and use the Web pages [24]. As it is known clearly that accessibility is, at least, as important as usability, many developers and designers didn't have an academic formation that gives them a deep understanding about accessibility and how to identify and fix accessibility problems. Actually they don't feel confident enough to apply or to evaluate accessibility. In order to provide Web accessibility, they will probably use some guidelines and tools found on the Web itself. Guidelines review is one of the most used methods to evaluate Web accessibility. It is an inspection method that consists on checking if the evaluated Web page meets a set of guidelines and checkpoints [25]. Although the method seems simple, Web accessibility guidelines review has several issues and is not always easy to apply. Though there are so many tools are available to evaluate accessibility but still there is problem in choosing one of them. Sometime it really becomes so tedious job to decide that which tool is giving less false positive results. And most of the tools do not provide completeness to the user as each tool does not have all features as users want at same time. In this chapter we will go through various web accessibility guidelines and also discuss on evaluation tools.

3.1 Guidelines to Follow

In a general context, and according to Bergman and Johnson [26], providing accessibility means removing barriers that prevent people with disabilities from participating in substantial life activities, including the use of services, products, and information. For the Web, Henry states that accessibility means that people with disabilities can perceive, understand, navigate, interact and even contribute to the Web [24]. To help designers to achieve Web accessibility, so many guidelines have been proposed. The most adopted set of guidelines is the Web Content Accessibility Guidelines 1.0 (WCAG 1.0), developed by the Web Accessibility Initiative (WAI) from the World Wide Web Consortium (W3C). European Union is the adopters of the WCAG 1.0, so that European public Web sites are encouraged to meet the guidelines. Other governments, like the Brazilian, have developed its own set of accessibility guidelines, but they are strongly based on WCAG 1.0 to guide the (re)design process of its public Web sites [27]. Australian Human Rights and Equal Opportunity Commission also work on WCAG 1.0 as accessibility standards [28]. WCAG 1.0 is a set of 14 guidelines, and each guideline has at least one checkpoint, totaling 65 checkpoints. Also, each checkpoint has a priority level, which indicates the impact of it on accessibility:

- **Priority 1:** A Web content developer must satisfy this checkpoint. Otherwise, one or more groups will find it impossible to access information in the document. Satisfying this checkpoint is a basic requirement for some groups to be able to use Web documents.
- **Priority 2:** A Web content developer should satisfy this checkpoint. Otherwise, one or more groups will find it difficult to access information in the document. Satisfying this checkpoint will remove significant barriers to accessing Web documents.
- **Priority 3:** A Web content developer may address this Checkpoint. Otherwise, one or more groups will find it somewhat difficult to access information in the document. A Web content developer may address this checkpoint. Otherwise,

one or more groups will find it somewhat difficult to access information in the document. Satisfying this checkpoint will improve access to Web documents.

There are three level of conformance to any web documents depending on number of priorities it is satisfying.

- **Conformance Level “A”**: all Priority 1 checkpoints are satisfied.
- **Conformance Level “AA”**: all Priority 1 and 2 checkpoints are satisfied.
- **Conformance Level “AAA”**: all Priority 1, 2, and 3 checkpoints are satisfied.

Although W3c has released a new version of the WCAG, the WCAG 2.0 [29], it will take some time until this new version fully replaces the previous one, as most of Web accessibility tools don't support WCAG 2.0 yet.

3.2 An Accessibility Guide

In this section we will discuss various accessibility guidelines like WCAG 1.0, WCAG 2.0, and Section 508. WCAG 1.0 and WCAG 2.0 are directly working under W3C's Web Accessibility and Section 508 of the Rehabilitation Act. We will discuss checkpoints of each guideline one by one.

3.2.1 WCAG 1.0

- **Guideline 1** (Provide equivalent alternatives to auditory and visual content): This guideline emphasizes the importance of providing text equivalents of non-text content (images, pre-recorded audio, and video).
- **Guideline 2** (Don't rely on color alone): This guideline also helps to the people who can not differentiate between colors. Also ensures that text and graphics are understandable when viewed without color.
- **Guideline 3** (Use markup and style sheets and do so properly): To ensure this guideline mark up documents with the proper structural elements and Control

presentation with style sheets rather than with presentation elements and attributes.

- **Guideline 4** (Clarify natural language usage): To ensure this guideline use markup that facilitates pronunciation or interpretation of abbreviated or foreign text.
- **Guideline 5** (Create tables that transform gracefully): To ensure this it is compulsory to ensure that tables have necessary markup to be transformed by accessible browsers and other user agents.
- **Guideline 6** (Ensure that pages featuring new technologies transform gracefully): Please ensure that pages are accessible even when newer technologies are not supported or are turned off.
- **Guideline 7** (Ensure user control of time-sensitive content changes): For this ensure that moving, blinking, scrolling, or auto-updating objects or pages may be paused or stopped.
- **Guideline 8** (Ensure direct accessibility of embedded user interfaces): Make sure that the user interface follows principles of accessible design: device-independent access to functionality, keyboard operability, self-voicing, etc.
- **Guideline 9** (Design for device-independence): For this use features that enable activation of page elements via a variety of input devices.
- **Guideline 10** (Use interim solutions): For this use interim accessibility solutions so that assistive technologies and older browsers will operate correctly.
- **Guideline 11** (Use W3C technologies and guidelines): Use W3C technologies (according to specification) and follow accessibility guidelines. Where it is not possible to use a W3C technology, or doing so results in material that does not transform gracefully, provide an alternative version of the content that is accessible.

- **Guideline 12** (Provide context and orientation information): To achieve this guideline Provide context and orientation information to help users understand complex pages or elements.
- **Guideline 13** (Provide clear navigation mechanisms): For Provide clear and consistent navigation mechanisms, orientation information, navigation bars, a site map, etc. to increase the likelihood that a person will find what they are looking for at a site.
- **Guideline 14** (Ensure that documents are clear and simple): Make sure that documents are clear and simple so they may be more easily understood.

3.2.2 WCAG 2.0

WCAG 2.0 is the updated version of WCAG 1.0 and follows on four principles: perceivable, operable, understandable and robust. All WCAG 2.0 guidelines work on these principles only. We will see each one by one.

Principle 1: Perceivable - To rely on this principle Information and user interface components must be presentable to users in ways they can perceive. Following are some guidelines for above principle

- **Guideline 1.1** (Text Alternatives): Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, Braille, speech, symbols or simpler language.
- **Guideline 1.2** (Time-based Media): Provide alternatives for time-based media.
- **Guideline 1.3** (Adaptable): Create content that can be presented in different ways (for example simpler layout) without losing information or structure.
- **Guideline 1.4** (Distinguishable): Make it easier for users to see and hear content including separating foreground from background.

Principle 2: Operable - To rely on this principle User interface components and navigation must be operable.

- **Guideline 2.1** (Keyboard Accessible): Make all functionality available from a keyboard.
- **Guideline 2.2** (Enough Time): Provide users enough time to read and use content.
- **Guideline 2.3** (Seizures): Do not design content in a way that is known to cause seizures.
- **Guideline 2.4** (Navigable): Provide ways to help users navigate, find content, and determine where they are.

Principle 3: Understandable - To rely on this principle Information and the operation of user interface must be understandable.

- **Guideline 3.1** (Readable): Make text content readable and understandable.
- **Guideline 3.2** (Predictable): Make Web pages appear and operate in predictable ways.
- **Guideline 3.3** (Input Assistance): Help users avoid and correct mistakes.

Principle 4: Robust - Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

- **Guideline 4.1** (Compatible): Maximize compatibility with current and future user agents, including assistive technologies.

In general there are 12 Accessibility Guidelines in WCAG 2.0 [30] as specified in Table 3.1. These guidelines should be followed in coding phase to improve accessibility.

3.3 Web Accessibility Evaluation Tool

There are Several accessibility evaluation tools available with various features that may help a lot when evaluating Web accessibility.

Table 3.1: Guidelines from WCAG 2.0[30]

SR. NO.	GUIDELINES	DESCRIPTION
1.1	Text Alternative	Provide text alternative for any non-text content so that it can be changed into other forms.
1.2	Time-Base Media	Provide alternatives for time based media
1.3	Adaptable	Create content that can be presented in different way(for example simpler layout) without losing information or structure.
1.4	Distinguishable	Make it easier for users to see and hear contents including separating foreground from background.
2.1	Keyboard Accessibility	Make all functionality available from keyboard
2.2	Enough Time	Provide users enough time to read and use contents.
2.3	Seizures	Do not design content in a way that is known to cause seizures.
2.4	Navigable	Provide ways to help users, navigate, find contents and determine where they are.
3.1	Readable	Make text content readable and understandable
3.2	Predictable	Make web pages appear and operate in predictable ways.
3.3	Input Assistance	Help users avoid and correct mistakes.
4.1	Compatible	Maximize compatibility with current and future users

3.3.1 Introduction

There are many accessibility guidelines those have been published for Web designers such as WCAG, but they are not implemented fully in the development phase . The reason is manual checking against the many checklists of the guidelines takes too long and it is expensive too. For reducing the cost of checking the Web contents for accessibility, many evaluation tools have been developed. But there is no such evaluation tool exist that can extract all of the accessibility problems perfectly. In this section we will know more about accessibility evaluation tool, their types, their usage, preferred feature to select them.

3.3.2 Types of Evaluation Tools

Web Accessibility Evaluation tools can be classified as automatic and semi automatic. The automatic tools allow developers and designers to quickly evaluate websites for accessibility. For example, Achecker [31] is an automatic tool to evaluate HTML content for accessibility related problems. For any level of conformance (A, AA or AAA, according to WCAG 2.0), it produces a report of all accessibility problems with their corresponding suggestions for corrections but this tool cannot make suggestions to help developers to correct the code of server source page. Also this tool does not provide any interface allowing developers to ?x bugs directly in the code. A-Prompt [32] examines Web pages at client-side for barriers to accessibility. It performs automatic repairs when possible and also assists the author in manual repairs when there is necessity.

The semi-automatic tools allow developers in partially evaluation of websites for accessibility [33][34]. But these tools are unable to give any explicit conclusion regarding website that it will pass the checking process. Specifically we can say that they require further interventions and deeper knowledge of developers and designers. For example, using Vischeck [33], developers can see the display of text contents and image combination for blind people. Through this simulation, developers can manually detect the color-related errors and modify the source code for fixing them.

3.3.3 Dos by Evaluation Tool

Web accessibility evaluation tools are significantly able to reduce the time and effort needed in carrying out evaluations. Moreover if we use these tools carefully throughout whole designing, implementation and maintenance part of development then these tools can help their users in preventing accessibility barriers, reporting the encountered barriers means they can improve overall quality of websites.

There are following ways through which tools can assist users in evaluating websites for accessibility.

- Some tools can perform determining the conformance of Web sites for checking accessibility and that can be executed automatically.
- Assisting reviewers in performing accessibility checks which further need to be evaluated manually.

3.3.4 Don'ts by Evaluation Tool

There are so many accessibility checks who need human judgment and there must be human evaluation using various techniques. But there may be chance that some evaluation tool may cause misleading results or false results such as not identifying any incorrect code. And these results cannot be used to determine conformance level without operation by experience evaluators who actually know all the capability and limitation of tools in order to achieve accurate results. We must understand this thing that web accessibility evaluation tool cannot determine the accessibility of web sites, they can only assist in doing so.

3.3.5 Usage of Evaluation Tool

Web accessibility evaluation tools can serve different purposes depending on the expertise of the users and what checkpoints users want to evaluate. Following are some of the common characteristics of evaluation tools for supporting users to fulfill different tasks during an evaluation process. There are some tools available those provide more than one mode of operation. Usage of Evaluation tools can be seen as:

Generating report: Report Generating Evaluation tool are designed for evaluating multiple pages or complete websites with very less or no user interaction. This tool summarizes the results of this tool in the format of reports which further can be customized according to the needs of the user. These tools are very useful in quickly determining the conformance of website to checkpoints which can be evaluated automatically and remaining checkpoints that need to be evaluated manually.

Step by step Evaluation: Wizard-based evaluation tools guide users through some sequences of checks step by step. Sometimes these tools execute some of the accessibility checks automatically and prompt the users for manually evaluating the remaining checks.

In-page feedback: This type of evaluation tools inserts icons and markup into the code of the Web pages to display the results of automated accessibility checks and their corresponding location within the pages. Sometimes, some other types of icons are also inserted into the Web pages to assist the manual evaluation of checkpoints.

Page Transformation: Transformation tools modify the appearance of the Web sites to help in identification of conceptual design issues with regard to Web accessibility. For example, transformation tools may present the content of Web sites in text only or without color.

3.3.6 Features of Evaluation Tools

There are some features which helps user to compare and access the web accessibility tool according to their specific need. Sometimes tools vendor mention additional information about how their tools support these or other features. Features are:

Accessibility: Accessibility of evaluation tool is equally important means people with disability must be able to access evaluation tool. Evaluation tool developers and

vendors can provide accessibility in different parts of the tools like user interface, documentation or generated reports.

Checkpoints Coverage: Most of the tools support broad variety of checkpoints but some tools focus on specific checkpoints.

Configuration: It is all about adoption of evaluation tool to the requirements of user.

Integration: It tells about how does the evaluation tool integrate into the web development environment of the users ? To provide integration there are three points that must be keep under considerations.

- **Platform support:** It must be ensured that the required tool features are supported on the platform where this tool will be deployed.
- **Software extension:** some evaluation tools are inserted as an add-ons.
- **Data Support:** Some evaluation tools can save data in to data base into specific format of file.

Policy Requirements: Some evaluation tools provide several accessibility guidelines as well as national policy requirements.

Reliability: An inaccurate result like not detecting the violation occurred decreases the credibility of a tool. Currently there is no widely accepted method available to determine the reliability of evaluation tool. So care full assessment of tool according to the specific need is the best way to measure its reliability.

Repair: Though it is not the part of evaluation process but it is next logical step. We provide Correction mechanism in our tool. In this we are providing to the repairing to the violated page.

3.3.7 Famous Known Tools

There are lots of Web Accessibility Evaluation Tools are available in the market. And in this section we are going to discuss about some very famous tools.

Achecker: It is an open source accessibility evaluation tool developed in 2009 by the Inclusive Design Research Centre, University of Toronto. This tool helps user to submit a web page via its URL or by uploading its HTML file and can subsequently select which guidelines to evaluate it against, namely the HTML Validator , Section 508 , Stanca Act, WCAG 1.0 and WCAG 2.0, BITV.

EvalAccess: It was developed by the University of the Basque Country in Spain. EvalAccess is one of the few tools that allow you evaluate an entire website for WCAG 1.0 compliance. It displays the results in an easy-to-read report, and describing each error detected. It is not user friendly tool and it is used for most of the designers and developers to clean up their sites.

WebAnywhere: It is a browser-based web application that works exactly similarly to screen readers such as JAWS. It's a very good way to see how a web page is read and navigated by assistive technologies.

Vischeck: Vischeck simulate the fact how a particular web page or image will look if it will be viewed by color blindness people. It simulates three types of color-blindness by entering URL or uploading image.

Wave: It is known as Wave accessibility versatile evaluator. This tool was developed by WebAIM and it is available both online and as a Firefox add-on. It reports accessibility violations by generating copy of the page that was evaluated and at the same time, providing guidelines on how to make correct them. It does not provide a complex technical report but shows the original Web page with some embedded icons and indicators that depicts the accessibility information within your page. Wave is motivation to our new Tool DCWA.

3.3.8 New About DCWA

DCWA is a new Designed tool . Though it checks accessibility in web contents similar to other tools explained in details in above section but still it is better than other tools. It provides checking as well as correction mechanism to accessibility guidelines violated page. And this extra feature of detection as well as correction makes this tool better from other accessibility tools. Detail discussion about the tool is given in next chapters.

Chapter 4

Detection & Correction In Web Accessibility

DCWA is an add-on in automatic evaluation tool for finding and correcting web accessibility issues in a webpage or website and that will help web developers to provide solutions for web accessibility issues.

4.1 Motivation

Nowadays Web-designers have focused on designing the interface that is eye pleasant due to which the issues of web accessibility have been poorly addressing and it becomes problem because users do not have the equivalent ability to accesses and exploit the web content. Therefore several guidelines and recommendations have been introduced so that web content can be made more accessible and usable to all users.

There are too many guidelines to follow which leads to the invention of accessibility checking tools such as automatic [20][31][32] and semi-automatic ones [33][34]. And these tools play very important role in dealing with web accessibility issues. But it was seen that a large number of tools those are available in market are just providing the features of detecting the issues of accessibility in a web page or we can say that they are lagging in Correcting mechanism simultaneously.

More precisely the fact is that the tools detect the issues and report it to developers

in the form of tips or complete documentation and now its developers or designers responsibility to deal with issues or make them correct manually.

This becomes motivation to make such automatic tool that can effectively provide both detection and correction mechanism simultaneously.

4.2 Description of System

DCWA supports the accessibility analysis and also facilitates fixing of the identified problems. A process for the identification and fixing of accessibility guidelines violations is proposed, together with a tool supporting the execution of the process.

System with three phases parsing, detection & correction are working separately with individual input and corresponding output that solve the problem of manually fixing the bugs of reported accessibility issues.

Here Parser reaches to URL and not only download the webpage mentioned but also do something especially useful for example extract particular parts of data on the web page, download selected files, and perform various other actions. It parses HTML tags and extracts data that further is being used for accessibility evaluation. This is very important and initial phase of evaluation. In general it can be said that Parsing is to divide a string into tokens based on the given delimiters. In market there are so many parser or say web parsers are available those can serve this function easily. This system uses StringTokenizer class which provides the first step in parsing process. It is often called lexer (lexical analyzer) or scanner. So with the help of this tool individual tag of any html file can be dealt. Simply it helps to access the tags individually and detecting the accessibility issues in them.

Detection is second important phase of the tool. There are certain guidelines to make a webpage accessible but web page that does not follow those guidelines do accessibility violation. Detection part of this system will go to individual tags and find out whether it contains any accessibility issues or not. If yes so embed some special symbol to show the issues are detected. This is important among all the 3 phases because in this phase issues are actually taking care of. Sometime same tool can not be used for detecting

accessibility issues in all webpage because most of the evaluation tools follow domain specific accessibility guidelines. But here main concern is to provide such tool that can simultaneously detect and correct web inaccessible webpage.

Correction is the third important part of DCWA. In Detection part accessibility issues were detected And correction part will provide correction mechanism to the corresponding detected issue. Also corrected page can be easily be used by screen reader as well as fully accessible software or system.

Detailed Discussion about each phase is given in next sections.

4.3 Parsing

Parsing provides splitting a sequence of characters or values into smaller parts. This is basically used for recognizing characters or values that occur in a specific order.

4.3.1 Parsing web page

Firstly, web page is parsed means download it and extract interested features say tags.From above description it is clear that tool is taking the use of StringTokenizer class to parse string from html file that will extract each element one by one and form the token and those tokens can be further used to find accessibility issues. This will be covered in next section Detection. Below is the figure that gives a clear picture about parsing.

To use StringTokenizer,specify an input string and a string that contains delimiters. Delimiters are characters that separate tokens. Here spaces are taken as a delimiter that actually gives complete word or group of words separated by space and space is a by default delimiter. Pickup each string one by one and parse it to parser that will further generate desired token for that can be further utilized for detection and correction purpose. StringTokenizer is a class in java.util package. Mainly there are two methods to solve the purpose of extracting tokens.

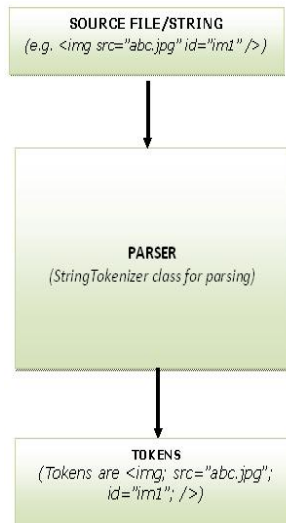


Figure 4.1: parsing source file

hasMoreTokens(): This gives true if more tokens are available, otherwise fails.

nextTokens(): This gives the next token available in the string.

4.3.2 Results

The results of parsing is tokens generated that are further going to be used as core input to detector for detecting accessibility issues. For example if we have `` as a string to be parse in parser that is using StringTokenizer so we will get tokens as using space as a delimiter. Now we can perform string matching to received string. And we can find all image tags being used in whole file. In our examples this image tag does not contain alternative text so it is an accessibility issue and we have to take care of this. This issue will be detected in detection phase and will be corrected in correction phase. In this manner we can work on individual tags like anchor tag, form tag or many more.

4.4 Detection

Detection mechanism searches for violation of predefined guidelines in client page or say html page. A violation is identified and reported according WCAG techniques.

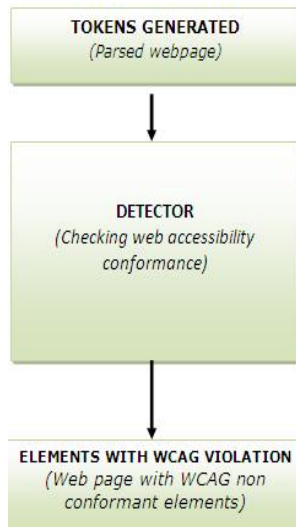


Figure 4.2: Detection

e.g. Missing alternative text, missing attribute, missing of text contents or missing of sub-tags of a specific code.

4.4.1 Detecting web Accessibility Issues

Detecting or checking web accessibility issues in a already parsed web page is very important in proper detection. Here Detector checks web accessibility conformance in particular element. Figure 4.2 explain this.

Detector detects web accessibility issues in a web page by simply checking WCAG violation and then and then specify violation source that specify missing features that actually causes violation.

4.4.2 Result

Result of this Detection procedure is to report the identified violation according to WCAG technique. Also to generalize our structure we call **victim element** to all those elements that have some set of violations. And we can categorize this violation as:

- **Violation Code:** It describes the violation in a code according to WCAG guidelines.

- **Element type:** There are three types of elements that can be found as not present in the source code of HTML. These are missing attribute, missing of text content or missing sub-tags of specified tag.
- **Missing source:** It specifies missing features that causes violation.
- **Reporting:** It illustrates the way to fix the victim element and to fix the code.

Once All victim elements are reported and suggestion about how to fix it comes use these weather manually or by an interface that will provide correction mechanism. In above example image tag does not contain alternative text and this is an accessibility issue that has to be resolved. Resolution to the issue will be given correction phase.

Most of the evaluation tools available in market do not provide interface allowing developers to bugs directly in the code. e.g. IDI web accessibility checker, Vischeck, Link Context Checker, Wami toolkit are some good evaluation tool those do not provide any interface to correct code. But this tool DCWA provides an interface for correction. That makes it better in this respect compared to other some famous and good tools. Now correction mechanism will be discussed in section 4.5.

4.5 Correction

Correction Mechanism can allow developers to modify the violation detected source page and also allow modification to code manually.

4.5.1 Correction Based on Detected Issues

Last section discussed about detection mechanism that is detecting victim elements and also provision of interface that allow modification in code level of source page. So that modification comes under correction mechanism because we are correcting code to make it free from WCAG violations. Figure 4.3 gives the clear explanation of correction mechanism.

In Figure 4.3 mechanism of correction procedure is clearly explained. victim elements detected pages with some additional option in the form of navigational image to

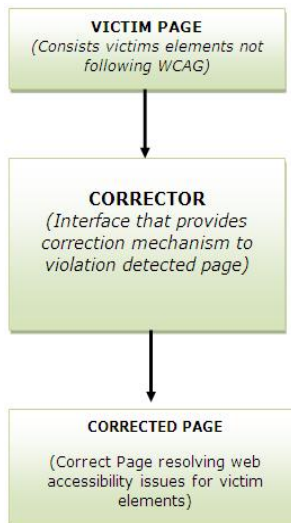


Figure 4.3: Correction of WCAG violation detected page

open correction interface of corrector through which modification of page can be done on the basis of WCAG techniques.

4.5.2 Result

Corrector gives modified pages following some or all WCAG techniques. It is always very difficult to include all the guidelines in our tool because some guidelines are compulsory to be followed to make it accessible but some are totally dependent on developers. For these types of guidelines detector generates suggestions for manually correction to make page accessible. In above example image tag was not containing alternative text. In parsing it was treated as a token then in detection phase image tag becomes a victim element violating the accessibility guideline. Finally correction to this violation is provided in correction phase. Developer can add Alt, attribute of image tag, as an alternative text from java applet as an interface available on clicking error symbol in detected phase. And that text will be informative rather than adding by default text. And that full fills the requirement of availability and usability of particular attribute.

4.6 Process Workflow of Tool

The whole process can be easily understood with the help of process work flow of DCWA tool, Figure 4.4. DCWA tool takes source page as an input and it goes to parser first that tokenize it into different tokens. These tokens go to Detector where violation of WCAG is checked and tokens are marked as victim element if there is any violation. Finally these victim elements go to corrector for correctness of accessibility issues. And some issues are solved manually on the basis of detector suggestion.

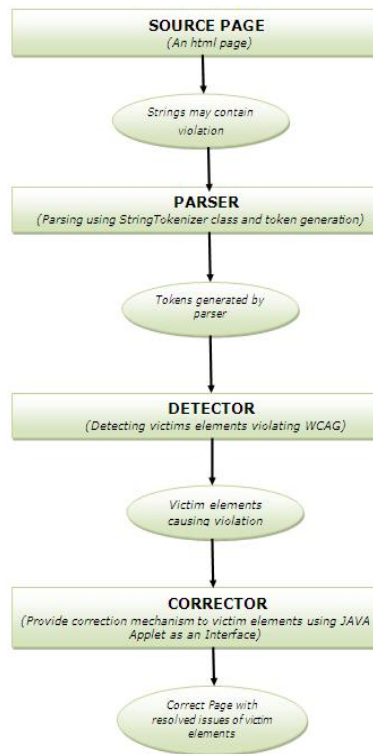


Figure 4.4: Process workflow of DCWA

Chapter 5

A Delivery Model for Accessibility Evaluation

This chapter presents working model for tool DCWA that will explain execution of tool under various phases.

5.1 Demo Of Tool

DCWA is stand alone application packaged in to jar. And to Run this tool. To run the tool just click on above icon. Also it can run on any system having java runtime environment.



Figure 5.1: [click here to run tool](#)

This figure shows loading of tool and instructions to wait for sometime. This window gives message of choosing file.

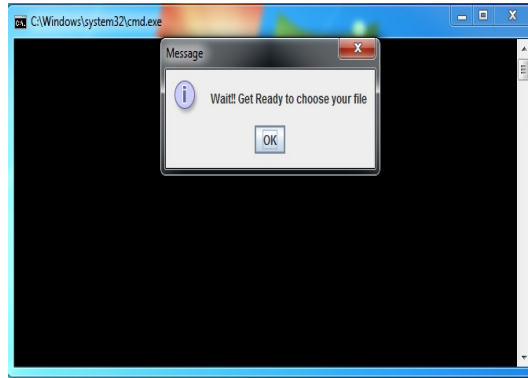


Figure 5.2: Loading of tool and instruction to choose a Page

This window shows browsing of file . Choose your file from system. choose input.html file.

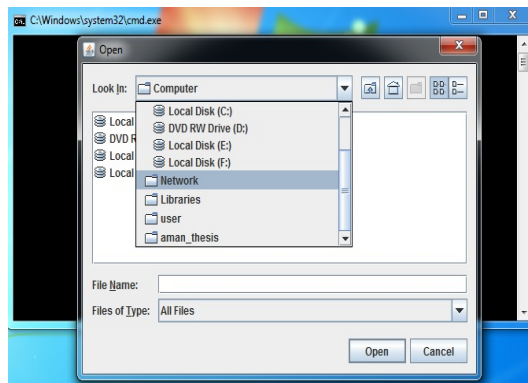
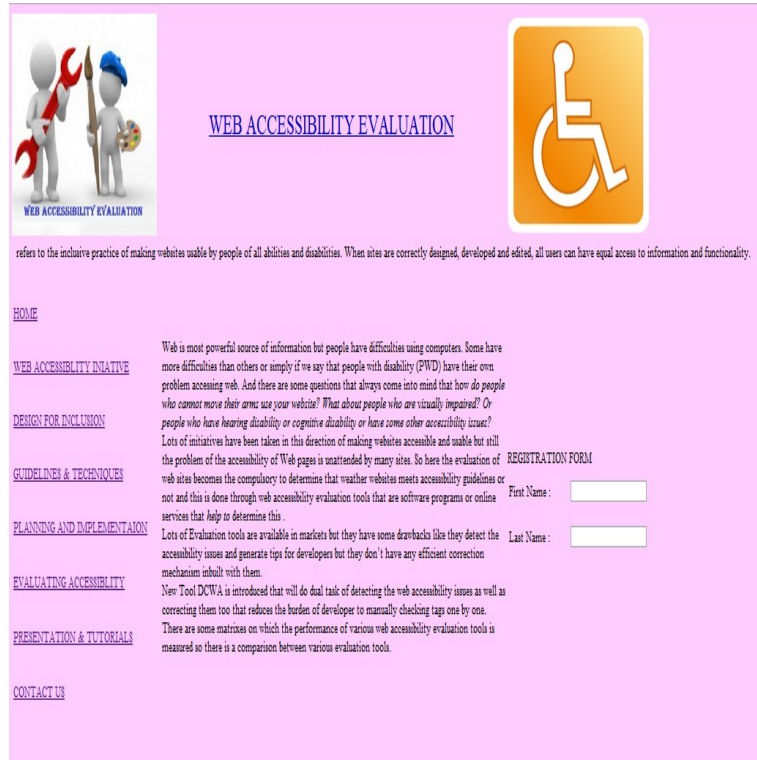


Figure 5.3: Browse Your Page

This is input.html file which is under observation. This input.html will go for parsing in parser.



The screenshot shows a web page with a pink background. At the top left is an image of two 3D figures, one holding a red ribbon and the other a paint palette, with the text 'WEB ACCESSIBILITY EVALUATION' below it. At the top center is the title 'WEB ACCESSIBILITY EVALUATION' in blue underlined text. At the top right is a white wheelchair icon on an orange square background. Below the title is a paragraph: 'refers to the inclusive practice of making websites usable by people of all abilities and disabilities. When sites are correctly designed, developed and edited, all users can have equal access to information and functionality.'

On the left side is a vertical navigation menu with the following links: [HOME](#), [WEB ACCESSIBILITY INIATIVE](#), [DESIGN FOR INCLUSION](#), [GUIDELINES & TECHNIQUES](#), [PLANNING AND IMPLEMENTATION](#), [EVALUATING ACCESSIBILITY](#), [PRESENTATION & TUTORIALS](#), and [CONTACT US](#).

The main content area contains several paragraphs of text. The first paragraph discusses the challenges of web accessibility for people with disabilities. The second paragraph mentions various initiatives and the need for evaluation. The third paragraph discusses evaluation tools and their drawbacks. The fourth paragraph introduces the DCWA tool. The fifth paragraph mentions matrices for comparing tools.

On the right side, there is a 'REGISTRATION FORM' with two input fields: 'First Name : ' and 'Last Name : '. The form is currently empty.

Figure 5.4: Input Page under observation

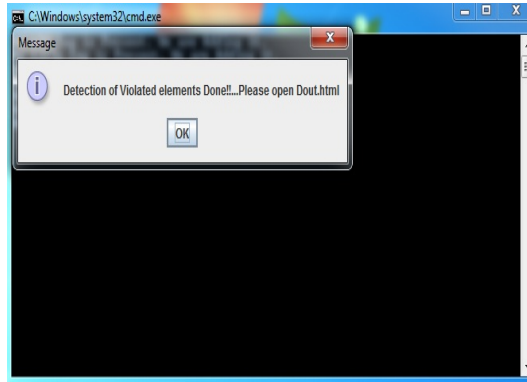


Figure 5.5: popup window showing generation of detection of file

Violation to WCAG guidelines in html file are detected.



Figure 5.6: Violation Detected Page with embedded icons showing victim elements

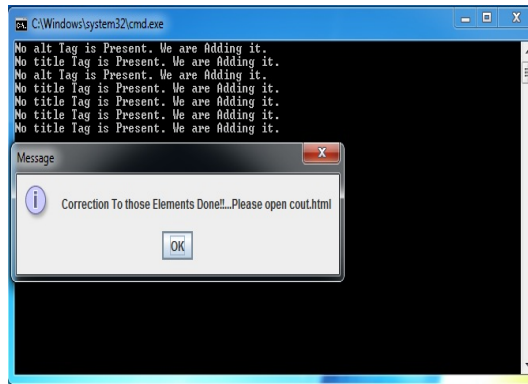


Figure 5.7: popup window showing that victim elements have been corrected
corrected file following wcag technique.



Figure 5.8: Accessibility Violation free Page showing correction mechanism

5.2 Proposed Approach

DCWA tool is using DCWA.java which is using StringTokenizer package. DCWA.java consists main approach.

DCWA.java

```
import java.io.*;
import java.util.StringTokenizer;
import javax.swing.*;
import javax.swing.JOptionPane;
public class DCWA
{
static String addTags(String str, String tag)
{
System.out.println("No "+tag+" Tag is Present. We are Adding it.");
StringTokenizer tok2 = new StringTokenizer(str);
String str3 = tok2.nextToken()+" "+tag+"="+" Added "+tag+" Tag by DCWA "+" ti-
tle= "+tag+" Added by DCWA " ";
while (tok2.hasMoreTokens())
{
str3+=tok2.nextToken()+" ";
}
return str3;
}

static String addTitleTags(String str, String tag)
{
System.out.println("No "+tag+" Tag is Present. We are Adding it.");
StringTokenizer tok2 = new StringTokenizer(str);
String str3 = tok2.nextToken()+" "+tag+"="+" Added "+tag+" Tag by DCWA " ";
while (tok2.hasMoreTokens())
{
```

```

str3+=tok2.nextToken()+" ";
}
return str3;
}
static String delmarquee()
{
String str3=" ";
return str3;
}

    public static String browseFile()
    {
    try
    {
JOptionPane.showMessageDialog(null,"WAIT!!!PRESS OK TO SELECT FILE TO BE
EVALUATED");
}
catch(Exception ex)
{
}
JFileChooser chooser = new JFileChooser();
chooser.showOpenDialog(null);
File f1 = chooser.getSelectedFile();
String str=f1.getName();
return str;
}
public static void main(String str[]) {

    String path= browseFile();
file f1 = new file(path);

```

```

filedf2 = newfile("C: \\Users\\aman\\Desktop\\aman.thesis\\MAIN_DCWA\\doutput.html");
Filecf2 = newFile("C: \\Users\\aman\\Desktop\\aman.thesis\\MAIN_DCWA\\coutput.html");
BufferedReader din;
PrintWriter dout,cout;
String str1="";
String str2="";
String cstr1="";
StringTokenizer tok1;
try
{

    din=new BufferedReader(new FileReader(f1));
    dout = new PrintWriter(new FileWriter(df2));
    cout = new PrintWriter(new FileWriter(cf2));
    while((str1=din.readLine())!=null)
    {

        tok1 =new StringTokenizer(str1);
        cstr1=str1;
        while (tok1.hasMoreTokens())
        {
            str2=tok1.nextToken();

            //System.out.println(tok1.nextToken());
            if(" < img" .equals(str2)&&str1.indexOf("alt") == -1)
            {
                str1="< img src="+ " "images/skull.png " title= " Erroor: alt is missing " alt= " alt
                is missing " "+ " >< /img > " + str1;
                cstr1=addTags(cstr1,"alt");
            }
        }
    }
}

```

```

    }

    elseif(" < a ".equals(str2) && str1.indexOf("title")==-1)
    {
    str1="< img src="+ "images/skull.png" title="Error: alt is missing" alt="alt is missing" + "><
    /img >" +str1;
    cstr1=addTags(cstr1,"title");

    }

    else if(" !marquee".equals(str2))
    {
    str1="< img src="+ "images/greenskull.png" title="Error: Marquee is present" alt="marquee
    is
    present" + ">< /img >" +str1;
    cstr1=delmarquee();
    }

    if(" < head >" .equals(str2))
    {
    System.out.println("hii aman jain");
    }
    else if("< head >".equals(str2)&& str1.indexOf("< title >")==-1)
    {
    System.out.println(str2);
    str1="< img src="+ "/"images/skull.png/" + ">< /img >" +addTags(str1,"< title >");
    }
    else if("< form".equals(str) && str1.indexOf("lable"))
    }
    dout.println(str1);

```

```

cout.println(cstr1);
}
din.close();
dout.close();
cout.close();
JOptionPane.showMessageDialog(null,"DETECTION OF VIOLATION ELEMENTS
DONE!! PLEASE OPEN DOUT HTML");
Thread.sleep(1000);
JOptionPane.showMessageDialog(null,"CORRECTION FOR VIOLATION ELEMENTS
DONE!! PLEASE OPEN COUT HTML");
}
catch (Exception e)
{
e.printStackTrace();
}
}
}
}

```

DCWA.java consists proposed approach of tool DCWA. It is using three packages input- output, StringTokenizer and swing packages. StringTokenizer packages is helping in parsing of input file that is further goes to detection of violated elements and dout.html file is generated. At last dout.html file goes to correction mechanism that generates cout.html.

Chapter 6

Conclusions And Open Research Lines

DCWA helped developers and designers to check their web pages for any accessibility issues. It is better than other available tools because this tool provides both detection as well as correction mechanism simultaneously. Though these facilities are available but they are separate. This tool is a new approach in evaluation of web site.

In today's world where use of web is increasing day by day and there are so many scripting languages in use. And they are used at server side. Any tool must provide mapping between server side as well as client side page.

REFERENCES

- [1] “who invented world wide web.” [Online]. Available: <http://www.boutell.com/newfaq/history/inventedweb.html>
- [2] “Hyper text transfer protocol.” [Online]. Available: <http://www.w3.org/Protocols/>
- [3] “Uniform resource locator.” [Online]. Available: <http://www.w3.org/Addressing/>
- [4] J. Seale, “Disability and e-learning in higher education: Accessibility theory and practice.” 2006. [Online]. Available: <http://www.ascilite.org.au/ajet/ajet22/seale.html>
- [5] J. Baldwin, “The baldwin report,” Master’s thesis, University of Saskatchewan Saskatoon, 2000.
- [6] “Web accessibility in mind.” [Online]. Available: <http://www.webaim.org/intro/>
- [7] “Policies relating to web accessibility.” [Online]. Available: <http://www.w3.org/WAI/Policy/>
- [8] “Uncrpd: Un convention on the rights of person with disabilities.” [Online]. Available: <http://uncrpd.nilehsingit.org/internation-documentsdisa/disability-law-and-disability-acts-of-other-countries>.
- [9] M. H. Rioux, “Disability: the place of judgement in a world of fact*,” *Journal of Intellectual Disability Research*, vol. 41, pp. 102–111, 1997.

- [10] J. P. Bigham, “Intelligent interfaces enabling blind web users to build accessibility into the web,” Ph.D. dissertation, University of Washington, Seattle, WA, USA, 2009, aAI3370463.
- [11] S. Harper and Y. Yesilada, *Web Accessibility: A Foundation for Research*, 1st ed., ser. Human-Computer Interaction Series. London: Springer, September 2008, vol. 1, no. 1.
- [12] D. Sloan, “The effectiveness of the web accessibility audit as a motivational and educational tool in inclusive web design.” Master’s thesis, University of Dundee, 2008.
- [13] O. Kheir., “The accessibility and usability of websites: Relationships between measures from users, experts and guidelines.” Master’s thesis, The University of York, 2006.
- [14] M. Huenerfauth, “Generating american sign language classifier predicates for english-to-asl machine translation.” Master’s thesis, University of Pennsylvania, 2006.
- [15] S. Hollier, “A study into the impact of computing and internet-related technologies on people who are blind or vision impaired.” Master’s thesis, Curtin University of Technology, 2007.
- [16] J. A. Smith, “Developing web accessibility: Section 508 compliance of post-secondary educational web site home pages,” Master’s thesis, Capella University, 2009.
- [17] L. Moreno, “Awa, methodological framework in the accessibility domain for web application development,” Master’s thesis, Carlos III University of Madrid, March 2010.
- [18] E. H. Tanaka and H. V. Da Rocha, “Evaluation of web accessibility tools,” in *Proceedings of the 10th Brazilian Symposium on on Human Factors in Computing*

Systems and the 5th Latin American Conference on Human-Computer Interaction.
Brazilian Computer Society, 2011.

- [19] V.-H. P. M. Thi-Huong-Giang, Dat Trinh Tuan, “Checking and correcting the source code of web pages for accessibility,” in *An International Conference paper in IEEE*, 2012.
- [20] N. Fernandes, R. Lopes, and L. Carrigo, “On web accessibility evaluation environments,” in *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*. New York, NY, USA: ACM, 2011.
- [21] C. Bailey and E. Pearson, “An educational tool to support the accessibility evaluation process,” in *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*. New York, NY, USA: ACM, 2010.
- [22] T. Oikonomou, N. Kaklanis, K. Votis, G.-E. Kastori, N. Partarakis, and D. Tzouvaras, “Waat: personalised web accessibility evaluation tool,” in *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, 2011.
- [23] E. Pearson, C. Bailey, and S. Green, “A tool to support the web accessibility evaluation process for novices,” in *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, ser. ITiCSE ’11. ACM, 2011.
- [24] S. L. Henry, “Introduction to web accessibility,” 2005.
- [25] J. Lazar, *User-Centered Web Development*. Jones & Bartlett Publishers,, J. Lazar, Ed. Jones & Bartlett Learning, 2001.
- [26] E. Bergman and E. Johnson, “Toward accessible human-computer interaction,” in *Advances in human-computer interaction (vol. 5)*, J. Nielsen, Ed. Norwood, NJ, USA: Ablex Publishing Corp., 1995, pp. 87–113.
- [27] G. Eletrnico, “emag, acessibilidade de governo eletrnico modelo de acessibilidade,” 2005.

- [28] HREOC, “World wide web access disability discrimination act advisory notes,” 2002.
- [29] M. R. L. G. V. G. Caldwell, B.; Cooper, “Web content accessibility guidelines 2.0.” 2008. [Online]. Available: <http://www.w3.org/TR/WCAG20/>
- [30] “Web content accessibility guidelines web site.” [Online]. Available: <http://www.w3.org/WAI/intro/wcag.php>
- [31] “web accessibility checker,” 2011. [Online]. Available: <http://achecker.ca/checker/index.php/>
- [32] “A-prompt , a web accessibility verifier.” [Online]. Available: <http://www.aprompt.ca/>
- [33] “Vischeck.” [Online]. Available: <http://www.vischeck.com/vischeck/vischeckURL.php>
- [34] “Link context checker.” [Online]. Available: <http://www.dur.ac.uk/its/web/accessibility/tools/linkcontext>