National College of Ireland

MSc in Web Technologies

Evaluating Web Accessibility for Blind Individuals

Author:
Mr. Pat Murphy

Supervisor:
Dr. Keith Maycock

A thesis submitted in fulfilment of the requirements for the degree of MSc in Web Technologies, in the School of Computing, at the National College of Ireland.

November 2013

Declaration of Authorship

I hereby certify, that this material which I now submit for assessment leading to the award of master of science in web technology is entirely my own work and has not been taken from the work of others save and to the extent that such work has been sited and acknowledged within the text of my work.

Signed:

Date:

Abstract

As the world's population becomes further dependant on the web for services and information, a large number of people with disabilities are disadvantaged.

What was once considered a luxury, has now become a means for total interaction with society. A great deal of tasks now performed on a daily basis by people require some access to a web environment. Consider the Olympic games held in London in 2012. Tickets for these were only available online. It is estimated by the world health organisation (who) that approximately 1 billion people are disabled, suggesting that they are no longer a niche sector of the population. Of these, 39 million are totally blind while a further 246 million have some sort of visual impairment. Web accessibility can be defined as the practice of making a web site accessible to all types of users. There have been a number of attempts to address this, from international guidelines and standards to local and national legislation and laws enacted in various countries throughout the world. However, a lot of these standards and guidelines are focused on insuring that government sites conform. There is very little to enforce accessibility on web sites that are not government related. The Web Assessment Accessibility Model (WAAM) software, was designed to address this problem specifically in relation to blind users. Data was collected on evaluated pages, showing what was found, what was changed and what would need direct human intervention. The measure for success was the amount of items changed, versus those identified as potential problems.

Three sets of data were analysed. Irish educational sites and a random set of 100 sites from Ireland and Spain were selected by a Google search. Spain is one of the few countries that did not have to migrate a lot of legacy systems to web environments. As

a result, it was thought, that their levels of accessibility should have been better than most. In each of these data sets, only the home page of each site was evaluated.

Acknowledgements

I would like to sincerely thank all the people who helped me with this project.

To Dr. Keith Maycock, my supervisor and course director, your help was greatly appreciated. Without your advice, I doubt I would have been able to fulfil the requirements of the dissertation as completely as I have.

To Jonathan Lambert from mathematics support, thanks for all your assistance with the representation of my data. Your help with the statistical analysis, graphs and table layouts was invaluable.

To Tim Lawless and Peter Fleming in the library, your help in searching and conversion of electronic material was invaluable.

To my wife Alison and my children Stephen and Ciara, thanks for all your support during my college experience. Thanks for encouraging me to take the course in the first place, and for putting up with me during the good and bad times.

Contents

List of Figures

List of Tables

Abbreviations

ARIA	**A**dvanced **R**ich **I**nternet **A**pplications
EIAO	**E**uropean **I**nternet **A**ccessability **O**bservatory
JAWS	**J**ob **A**ccess **W**ith **S**peech
NDA	**N**ational **D**isability **A**uthority
POUR	**P**erceivable **O**perable **U**nderstandable **R**obust
TAW	**T**est **A**ccessibilidad **W**eb
W3C	**W**orld **W**ide **W**eb **C**onsortium
WAAM	**W**eb **A**ssessment **A**ccessibility **M**odel
WAB	**W**eb **A**ccessibility **B**arrier
WCAG	**W**eb **C**onteny **A**ccessibility **G**uidlines
WHO	**W**orld **H**ealth **O**rganisation

For my wife Alison...

Chapter 1

Literature review

1.1 Introduction

As the economics of service and information provision drive more content exclusively to the web, a large number of people with disabilities are disadvantaged [1].

What was once considered a luxury, has now become a means for total interaction with society. A great deal of tasks now performed on a daily basis by people require some access to a web environment. Consider the Olympic games held in London in 2012. Tickets for these were only available online. Services such as Amazon, and EBay can only be accessed online. It is estimated by the world health organisation (who) that approximately 1 billion people are disabled, suggesting that they are no longer a niche sector of the population. Of these, 39 million are totally blind while a further 246 million have some sort of visual impairment. Over half the computer using population have some sort of age related disabilities. These people would also benefit from accessibility considerations. Not every disabled person requires assistive technology, but for those that do, accessibility is of vital importance. This is especially true for the blind user. Regardless of whether it is required or not, it is easy to implement with today's technology, and can give an enterprise an edge when it comes to a business environment. Accessibility has always been an issue for a web environment. However, if it is properly implemented, sites can be ranked higher in Google searches as a result. The only real adoption of standards and guidelines is at government level, where users are accessing educational and library web sites as well as public sector services. This review

concentrates on two types of study, those conducted with automated tools, and those conducted with human evaluators. However, the implementation of either will greatly depend on what the study intends to report on.

1.2 Studies conducted using automated tools

Parmanto et al [2] proposed a new metric for measuring the accessibility of web sites for disabled users. When this paper was written, the measure was to use a method that produced results showing absolute compliance or failure. The metric proposed in this paper was to quantify the amount of passes and failures rather than an absolute. The guidelines used were the WCAG1 from 1999. A web site was determined to be accessible or not by evaluating it against the checkpoints in the guidelines. This paper proposed, that web accessibility should be measured in degrees rather than absolutes. There is also a higher probability, that web sites will yield an inaccessible result, as the amount of pages grow. The metric was based on the WCAG1 guidelines, which can be checked using an automated tool. The paper states, that the metric would provide an estimate of web accessibility, while actual measurement would require some form of human intervention. The metric proposed was the web accessibility barrier (wab). Although it was not designed to be a real measure, it does have practical strengths. The comparison was made between using an automatic tool with wab, and performing the same tests using human intervention on 100 web sites with large numbers of pages on them. An automatic metric evaluation is objective, allowing web sites to be compared side by side. The wab metric does not check guidelines that are specifically designed for human intervention, such as colour being used to convey information. Wab was designed to give an overall score based on the priorities of WCAG1, and the amount of violations on each page, based on the potential violations. Take for example, a page with 500 images on it, 499 of which have alternative text. This could not be rated in the same way as a page with one image on it, and no alternative text for that image. By using standard metrics, both pages would be considered to be inaccessible, when clearly, they are not. The metric does not take into account the location of any accessible barriers that may be detected. For example, the entire web site could be accessible, but the home page may not. This would prevent the use of the site. It was noted, that the metric compared well when tested against automation. This paper puts forward the idea, that metrics can be used

to evaluate a web site, whether simple or complex. It also acknowledges, that human intervention is required in order to get a real estimation of what the accessibility is like. Compared to the "accessible or not" approach, it is a welcome step forward. It is based on the WCAG1 guidelines, which insures, to some extent, its validity. It has also been developed along with a tool called Kelvin, which is used to retrieve the data. The tool was also developed by the writers of this paper. This makes the entire package very useful and a valuable tool for web designers into the future.

Hacket et al [3] conducted a study which asked if the home page was enough to use when evaluating the accessibility of a web site. The study focused on 50 web sites, which were selected by use of Alexa.com. The selection was taken randomly from the top 500 English sites on April 10th 2007. Each site contained at least three levels. Each level of the web site was scored using Kelvin (mentioned above). The tool used an automated approach to determine the scores of each level. This was then used to see the correlation between the different levels in order to determine whether the home page was enough when evaluating a web site. The paper found that the home page was not enough to determine the web site's accessibility, although there was a correlation between the home page and level one. Using both these results gave a better indication as to whether the site was accessible or not. The paper did acknowledge, that there were limitations in the web sites tested, and suggested, that future studies should use a larger sample in order to prove or disprove the results. The paper did not indicate, that the web sites were cached in order to run the tests. While this study is perfectly valid when using an automatic tool, it makes no reference to the "judgement" of human interaction. There is no indicators to say exactly what the tool tested for in its pass over the web sites. Although these results were valuable, they lack the human interaction, that could add to the study at a later time. For example, the tool may have found that all images on the home pages and level one pages of the sites tested had alternative text. However, without some human interaction, there is no way to be sure that the description is valid. It is assumed, that the web sites were not cached before the tests began. The paper did not indicate this. There is also no reference to how long each test took, and whether or not they were all completed in a single day. If not, there is the possibility, that the pages could have been changed after their initial identification on April 10th 2007.

De Andres et al [4] conducted a study into the factors influencing the web accessibility of big listed firms. The study was based on the WCAG1 guidelines. It suggested, that the reasons for lack of accessibility could be grouped into three categories; financial, operational and the csr (corporate social responsibility) strategy of the firm. The study was conducted on companies who were quoted in the selected indexes of four different countries, the USA, France, Spain and Germany. The sites were checked for two years, 2007 and 2008. Only since 2005, were financial records comparable on web sites owing to a European directive. However, only the home page of each site was checked. The study assumes, that the home page is indicative of the rest of the site, given that it is the "gateway" to the rest of the site. The web accessibility barrier (wab) metric was used when evaluating the pages. The reason for this, is that although it is automatic, it scores each individual WCAG1 guideline. For example, all images should have alternative text. Using WCAG1 alone, if one image on a page has no alternative text, the page fails because of this. With wab, it gives an overall score for the page, noting for example that there are ten images on the page, two having no alternative text. However, it should be pointed out, that wab does not cover all the checkpoints of WCAG1, and as a result, it is not perfect. Two automatic tools were used to compile the results, (TAW and HERA). It was found in the study, that operational factors had the most influence when changing the strategy towards web accessibility. Web accessibility policies are aimed at increasing the efficiency of the firm. Of the four countries studied, Spain had a higher accessibility score. The study notes, that this was to be expected, given Spain's internet infrastructure, and the fact that its population had more elderly people than that of other European countries. In a country where the development of web sites occurred at a later stage (Spain), it was found, that the costs were lower. This was noted to be influenced by the fact that there were less legacy systems to migrate. The study also found, that American sites were more accessible than their European counterparts. It was noted, that this was probably due to the fact of legal action being a greater possibility in the U.S. than in Europe.

Sandhya [5] wrote a paper which discussed the evaluation of web accessibility by use

of screen reading software. The software used was a trial version of jaws (Job Access With Speech), available from the Freedom Scientific web site. The paper did not specify what version it was. The web sites tested were random. The paper focused on the use of technologies such as Ajax, Adobe's Flash and Microsoft's Silverlight. A number of observations were made which were general. No one web site was singled out in relation to accessibility. The paper found, that with the introduction of advanced rich internet application (aria) technologies, the web became more inaccessible. Pages can be refreshed at random, and screen readers may or may not be aware of the changes. If the software was aware of the changes, there were a number of ways in which it could react. There were various suggestions made as to how this could be addressed. All of them made sense, but there implementation would very much depend on the type of web application concerned. The paper uncovered common problems which exist in almost every web site. It highlighted the fact, that web accessibility while being talked about, is not really in anyone's focus.

Eyadat et al [6] Conducted a study in which 100 universities were tested to see if their web sites were accessible. The study notes, that for educational institutions, a fully accessible home page is of vital importance, and is possibly the most important page on the site. Everything else relating to education at a specific university will be accessed from the home page. If the home page is not accessible, it makes navigation to other features on the institution's web site unreachable, regardless of their accessibility. Of the 100 universities tested, only one had a home page that was error free. It was also noted, that the admissions page of an educational institute is almost as important as its home page. Potential students will visit this, in order to see what requirements they must fulfil before they can be accepted. There could be some sort of online exam which has to be taken. This, usually, will be accessible via the admissions page. Out of the 100 universities tested, not one had a fully accessible admissions page. Disability services pages were also tested, and only one university had a page that was accessible. Given the census statistics compiled in this study, it was noted, that students with a disability would be more likely to receive financial aid for third-level education. Based on this, the financial aid pages of the universities were tested. Not one of them was found to be accessible. These tests were conducted using the test accesibilidad web

(TAW) automatic tool. This tool is capable of producing a report for each web page, and issuing warnings where manual intervention is necessary. The study concluded, that all the universities tested, had accessibility issues, and in most cases, they were quite severe. This study was carried out in the U.S. In every study, there is a certain level of human interaction that is required to evaluate whether the results are as bad as they seem. However, it would seem, that the results were fairly comprehensive, given the organisations that were tested. If action were taken even on the automated results, it would improve accessibility.

Johari et al [7] produced a paper which looked at the accessibility of the web via mobile devices. The mobile device area is vast, and changing almost on a daily basis. It noted, that most tools are used to evaluate web sites and not actual access. It was also noted, that the actual device itself could be the reason for lack of access despite the manufacturer's efforts to address this. Some applications were designed by the authors using open source tool kits, but little reference was made to them, or what they were capable of doing or providing. This paper mainly focused on what was already available in the form of legislation (section 508 in the u.s.) and standards, and what they should do. No one device was analysed, although a number of manufacturers were mentioned such as Nokia and Apple. There is assistive technology for Nokia phones, but it is extremely limited, and only works on certain models. The Apple iPhone has a built-in voice activated application called Ceri. However, it falls very short of what is required. For example, having instructed the phone to send a text message, the speech software is incapable of reading back the actual text. It does inform the user that a message is ready to send, but it cannot say what the message is, or where it is going. The paper was basically an overview of what was available, and what should be done.

Goncalves et al [8] conducted a study into enterprise web accessibility amongst the Forbes top 250 companies of 2009. This study was done using the automated tool SortSite, for compliance with WCAG, WCAG2 and the u.s. Section 508 guidelines. In general, it was found that enterprise level sites required significant improvement. This study argues, that large corporations should be leaders in this market, encouraging other

smaller companies with less financial facilities to follow their lead in as much as is possible. It notes, that if these companies do not comply with standards and legislation, then there is no encouragement for those smaller organisations to do so and with technology playing such a vital role in daily society, these larger companies should be more open to the possibility of increasing their share of the service and business markets by making their facilities more accessible to all. The SortSite tool was chosen, as the authors of this study believed, that it delivered a fully automatic evaluation of an entire web site, in a format that is easy to understand and interpret. The study recognised, that a human evaluation should also be carried out, but this was not done because of resource limitations. The companies chosen for this study were worldwide, and it was believed, that they represented a good control group of those currently in large business and services. Of the 250 sites chosen, only 94% of these could be evaluated. This was caused by incompatibility between the web sites and the tool being used. Some of this was caused directly by tools such as JavaScript and Flash. It was realised, that these tools have accessibility problems, and these have been passed on to the tools trying to evaluate them. Having analysed individual results, several abnormal values were discovered, which in the author's opinion, could compromise the entire sample. What these values were is not clear. However, it was decided to "statistically treat" the results, in their own words. The reasons for these errors could be measurement or execution errors. This in their opinion lead to a more reliable set of results. It was noted, that there is a much greater need for business and i.t. people to be aware of how accessibility should be integrated into these large sites. This study is very interesting, in that it targets a larger enterprise organisations. However, like many studies before it, it relies totally on automation for its statistics. This was acknowledged in the paper as a lack of resources. The paper does make reference to having to "statistically treat" the information before a reliable set of results could be obtained. It only mentioned possible things that could have produced the abnormal values when evaluating the various web sites. It was not clear, as to whether these sites were archived before being analysed. If not, this could account for the "abnormal values" discovered while obtaining the results. It also opens the possibility, that at the time of evaluation, the site could have been going through an update, or some other sort of change. It is not surprising, that these large companies have such a large volume of errors, even if these are being reported by an automatic tool. There is lots of legislation, guidelines and standards. However, there is very little law, and where there is, the company only needs to operate their web site outside the

control of that particular country. There is actually no way to have large companies conform to any particular standard. Indeed, it could be argued, that they are gaining that much profit from those that can access their web sites, that there is no need to conform in order that a minority has access also. It is smaller companies that are more at risk from non-conformance.

Hanson et al [1] wrote a paper which conducted a study of approximately sixty government and "top sites" from the U.K. and the U.S. since the WCAG guidelines were introduced in 1999, until 2012 with the now valid WCAG2 standards. They categorised top sites as those with high traffic volume such as commercial, news, healthcare, and higher educational content. Only the main pages of the sites were analysed, as it was considered to be a good indicator of the rest of the site content. Archived versions of the pages were used, for previous years where available. The study focused on accessibility issues for these sites for the last thirteen years. 952 top sites and 231 government sites were initially selected. However the number of sites available for the complete study was 34 top sites and 10 government sites. It was noted, that government sites performed better in the overall study, showing more links to accessibility statements, and having accessibility links on their home pages (in links or alt text). The testing conducted was automatic. There does not appear to be any "human" evaluation of these pages. This study was conducted over 13 years. In that time, there has been a significant change in the way that web pages are structured. There has also been a major change in browser support, what they can do, and their versions. Html5 has also been introduced. A great deal of processing is now done on the client side, something that may not have been as prevalent in 1999. There was, at that time, a great focus on desk-top machines, and many of the major government and business applications were available only on mainframe computers. The web was still very much in its infancy. With greater availability on the client-side to render graphics, and the ability to convey a lot more information over the web, the structure of these sites would have changed considerably. This study may not have proved anything other than the fact that the home pages of these sites had marginally improved over the last thirteen years. There is nothing to suggest, that things like the introduction of JavaScript was taken into account when performing the study. As a result, it would be very difficult to draw any meaningful comparison based

on home pages from a small number of sites over 13 years, using only the home page, and without making reference to how they were updated and maintained.

1.3 Studies conducted using human evaluation

Bailey et al [9] conducted two studies into the current state of web accessibility. It found considerable differences between those who specialise in web accessibility and those charged with web maintenance, in the ways that accessibility is perceived. Both studies also revealed, that there was little awareness of accessibility issues, and commitment of resources to solve the problem in organisations. Study one, was designed to gather the opinions of web specialists. These specialists were either web consultants, or those charged in their organisations with web accessibility. The results from this study were used to design study two, which charted if and how web accessibility was implemented in organisations. This study focused on web maintainers, those charged with the maintenance and update of the site and content within the organisation. In the first study, of the sixty consultants targeted, only 21 responded. What was remarkable about this, was that there was eight different countries represented in the responses. Once study one was analysed, study two was approached. This consisted of a mass email campaign. To encourage participation, a free evaluation of a web site was offered. Of the 86 responses, 79 organisations fully completed the study, 74 of those were from u.k. local government, who's level of accessibility had been criticised in relation to this paper. The conclusions of both these studies found that web specialists who encountered web maintainers, found their perceptions of accessibility to be negative, despite having a good attitude towards their work. General awareness of accessibility in organisations was poor, leading to the belief, that as non technical people added web content, accessibility issues would be introduced. This project did not specifically test web sites, rather those who develop, run and maintain them. It was evident, that there was a severe lack of training provided with regards to accessibility. Moreover, there was still a huge lack of compliance with legislation and the laws of different countries. The problem is worldwide, and is not confined to any one sector. Add the availability of open-source tool kits such as JQueryUi and DreamWeaver, and the problems increase. There are a

number of web providers allowing the "ordinary Joe" to build their own web site. This process is designed to be easy, as no prior web experience is required. The cloud has introduced another dimension into this, in that nothing that is deployed is checked for accessibility by any cloud provider. However, if cloud providers were to provide some sort of validation testing before deployment, the problem could be partly addressed.

Bruyere et al [10] conducted a study into accessibility implications for the employment of people with disabilities. The study sited another paper, which found that 88% of h.r. departments surveyed used online facilities for recruitment. Another finding, was that a lot of firms allowed their employees access to the h.r. site in order to accomplish tasks such as the filling in of time cards, and flexi-time systems. The study found that employee training online was increasing. A lot of companies are also offering online support, with customers being able to ask for support via the web. Some firms have admitted, that they have trimmed their staff so much, that employees could not handle the work load without the aid of supporting technology. A review quoted in this paper found that only one-third of recruiting sites were accessible, and only one-quarter of job boards were accessible throughout the entire application process. The information for this paper was gathered via a 10-15 minute phone survey to 813 members of the largest h.r. organisation in the U.S. It gathered information about the extent of computer use in the organisation, and its use of online technology in relation to h.r. processes. Awareness of accessibility barriers was also explored in relation to people with disabilities, knowledge of assistive technology, familiarity with computer adaptations that may be required for employees, availability of employees trained in accessibility and the helpfulness of the organisation in relation to accessibility. Of the 813 members approached, 493 individuals were contacted, with 433 completed the survey. When asked about assistive technologies, the one that emerged as having the least exposure was screen readers. Over 70% of those surveyed were unfamiliar with them. The percentages were lower in relation to screen magnifiers, and other types of assistive technology. When asked about web site evaluation, 30% said that their sites were not evaluated for accessibility. The rest did not know. However, training was provided in the areas of accessibility, but this was not specifically related to online processes. Where internet accessibility training was given,

it was mostly i.t. staff that received it. Approximately 71% were interested in receiving further training in this area. A lot of those surveyed did say that they had made computer adaptations for disabled employees. When asked about resources within the organisation to help, the highest rated resource was the employee themselves. Although it was carried out in the U.S. this type of survey would yield the same results regardless of its location. There is legislation and guidelines to help with accessibility. However, they are not enforced, as there is no way to insure that this is done. As noted, the employee themselves was the highest rated resource when it came to making adaptations. Because of the lack of awareness, management staff are not interested in one or perhaps two employees when it comes to procurement of i.t. systems, or web sites. The reality is, that most of these facilities are either outsourced, or bought as packages. Because of their lack of commitment and lack of training, accessibility is the last metric considered when obtaining new i.t. technologies. This paper was written in 2006, some seven years after the WCAG1 guidelines were published. As the study shows, practically no attempt was made to interact with them.

Craven et al [11] conducted a task-based study for the European Internet accessibility observatory (EIAO), which set out to prove that statistical measures could be applied to an initial assessment of a web site. These could then be used in further assessments of the web site to keep track of its accessibility development. It found that by providing participants with a task rather than letting them randomly explore a web site, the results were more structured. This allowed for comparison between automated and user-driven results. The participants for the eiao user testing were selected based on their i.t. experience, disability, and use of assistive technology. Twenty people were selected, all having five or more years experience and were either blind, visually impaired, hearing impaired, physically impaired or had learning difficulties such as dyslexia. A control group of non-disabled people was also selected. Assistive technology such as Jaws(6.1, 6.2, 7.0), Supernova(6.5)(Screen readers), (Lunar 5.21), (zoomtext 7.1) (screen magnification), (dragon naturally speaking) (voice recognition) and pc settings to change text colour were used. Tasks were prepared for the test, and web sites were selected. The users were allowed to explore the web site, and then asked to find and describe an image

on the page. This task was considered to be useful, as it could be evaluated using automated tools as well as user feedback. A number of other tasks and automated tests were also developed and used in this study. Web sites used were cached in order to maintain consistency over the testing phase. The study was also conducted remotely, using an online form. This type of testing brings in both the human and automated forms of testing. It was noted, that neither of these can be completely relied upon. They each have their strengths and weaknesses. There are good reasons for using either, depending on the question being asked, or the problem to be solved. Remote testing however, has its own problems. Apart from the versions of software being used, there is no reference made to what else was running on the user's computers at the time. This would refer to the version of windows, the browser being used, and their current update level. All these have an effect on how and when information is presented, especially when it comes to screen readers. Screen reading software can also be effected by the type of hardware being used. With older versions of jaws for example, there were certain types of monitor that were not supported. Jaws version 6, did not support high definition television screens, when the computer was connected. Some audio video cards were also not supported, or gave problematic results. The face to face study is much more controllable, and yields a better level of information. Users will tend to write down on forms, only as much as they need to. A controlled environment will allow the users to express what it is that they are feeling at the time. It may also be useful for the project co-ordinators to actually "see" what the blind user does not. In the task mentioned above, the participants were asked to look at an image on the page, and evaluate it. This was only one of the tasks required. Take for example, the jaws screen reading package. In a remote environment, the user could have graphics turned off, or set to "labelled graphics". If the image had no alternative text, the user would not be aware that it was even on the page. However, if graphics were set to "all graphics" it would be spoken regardless of alternative text. This could be the difference between the user saying that they could not find the image, or that it was found, but had no text attached. A controlled environment would reveal this, without any problems, as all the computers would be set up to reflect the test environment. Using different versions of assistive technology also brings in a degree of variability into the study. New versions have new features. It is possible therefore, that jaws version 6 did not recognise something that jaws version 7 did. This could also be stated for the other software being used in the study. The only thing in this study that was completely stable was the web sites used. As stated in the paper, they were cached,

to prevent them from change while the study was being conducted.

Karreman et al [12] conducted a study based on accessible web site content guidelines for people with intellectual and language disabilities. The study was based on two versions of the same web site. One of these was made "easy to read", while the other had no changes. Both of these sites were tested on two groups of twenty users. One group had intellectual disabilities, but could read, while the other group had no identified disabilities. The results proved, that the group with disabilities found it much easier to access the easy to read site, while there was no discernible difference in productivity with either site when tested with the other group. The focus of the study was on the guidelines regarding the verbal content of the web site. The study also focused on how usable the site was, in that no one group's needs should take from the usability of the site in relation to another group. Both groups were divided evenly. Ten from each group used the adapted and non adapted site. Satisfaction with the web site was measured on a scale from one to five, (one being definitely disagree to five being definitely agree). Questions were asked of each participant after they had completed their tasks. Some of the answers could be found within the web site's content, whereas, other questions were multiple choice. The participants carried out the tasks in their homes. The sites were presented to them on a laptop, which was brought by the person conducting the experiment. The experimenter was aware of which users had and had not got intellectual disabilities, and the paper freely admits, that this may in some cases, have influenced the amount of assistance given when absolutely required. Remarkably, the study found that more assistance was required by users accessing the adapted site. One of the conclusions from the study, was that the participants without disabilities preferred the non adapted site, whereas, the other group preferred the adapted site. The recommendation was made, that adapted sites for those with intellectual disabilities should be used, but that all visitors should be notified about this perhaps on the home page, explaining why it has been done. This study seems to have been conducted in a very controlled manner. Unlike others, the content and versions of software being used was tightly controlled. As mentioned above, the sites were presented to the users in their homes by the use of a laptop. This would insure, that the content of the site was consistent. However, no mention was made of whether assistive technology was used for either site. The study focused on those people

with intellectual disabilities, but there are those in such a group who would perhaps require assistive technology in order to help with language difficulties. This was not made clear. In order to maintain this type of environment, two different versions of the web site would need to be maintained. This is not always possible, or a good idea. With the emergence of new technologies, it may be ultimately easier and more productive, to conduct a similar study on a web site that has been built with intellectual disabilities in mind. This site would be a single site, with the ability to adapt when required. It is ultimately easier to maintain such a site with regards to its content. There will also be very few web sites that have an "adapted" version for any disability. The usual way is to construct the site with accessibility in mind when developing it. This assumes, that accessibility has been thought about at all. To conduct a study on a specially built web site in itself, can give misleading results, as this will not be the norm. A majority of web sites will be "general" for want of a better word.

Debevc et al [13] conducted a study into the accessibility of the web for deaf users. The study uses the WCAG1 guidelines. Although other guidelines have been released by other organisations, it was felt, that they were too general to be of much help. Most of them require information on the web to be translated into some form of written text. It was noted, that most deaf people consider written text as their "second" language. A tool was developed as a plug-in to enable the delivery of transparent video to the user. It was noted, that standard video took over the screen, and had a distracting effect when used by the general public. The transparent video tool allowed the user to control the video in their own time. Once the video had been completed, the window was closed, returning the site to its standard look and feel. This removed the need for a double version of the same web site. It was very clear, that the use of transparent video increased the user's perception of the site and its content. Furthermore, the user was more interested in the site, and other sites using this type of technology. Transparent video allows the developers to use the existing design in place, and add interactive elements that trigger the videos when required. Statistics quoted in the study show that approximately 10% of the world's population have hearing difficulties, and that sign language is their primary form of communication. There were other projects referred to in this study that currently have video information displayed. However, it took up a

large proportion of the web page, and had to be very carefully planned. The WCAG1 guidelines allow for the use of clear and simple language where the language used on the web site is not the first language of the user. However, these guidelines do not cover the use of transparent video. This was provided by the use of flash and dhtml. The use of dhtml allowed the video to be treated as a separate layer on the page. Other projects were mentioned, where text already on the web page was translated into sign language. This had its drawbacks, as it could not always be accomplished. The idea here, was to provide a video that showed the page content, without translating what was currently being displayed to the standard user. The most popular approach is for web developers to use flash player to show current sign language videos, devoting part of the page to this specifically. The advantage of this approach is that it is cross-browser compatible, and there are reduced security issues. Obstacles include the fact that web sites frequently contain text, images and photos through the entire page, leaving very little space for the video frame. One of the solutions suggested is the use of a pop-up window. However, certain browsers do not allow these type of windows. It also has the effect of taking the user's focus from what is underneath the window. Another approach is to launch some type of player such as Windows Media Player. However, the user is now presented with two windows on screen, which requires their attention. It was noted, that regardless of the format used, hard-of-hearing users must have full control over the video, and that the video should be of sufficient quality as to allow the user to enlarge it without the quality suffering. A reason for this, is to allow the user to have a larger view of the facial expressions and the hand jesters being used. In the experiment, an upgrade of the web site was done, rather than a separate accessible version. The icons for the videos were inserted onto the page by use of html. All video control was performed using JavaScript code blocks. The videos were stored on a central server for retrieval. The results of the experiment concluded, that most deaf users would like to see transparent video included in most web sites. Only a small proportion of them concluded, that the video was over-sized. There were also some who felt, that transparent video would not be required on all web pages of a site. Overall, it was considered a good move forward with regards to accessibility for this particular group. Although this study was very conclusive, there could be issues which were not explored. It was suggested, that videos be loaded with the initial web page, and code markers used to activate it. There could be performance problems in relation to this. Users can become irritated, if a web page takes too long to load. This can be caused by scripts executing before the page is displayed. Usually, it

is preferable to have this done in the background after the user has received the page. However, in the case of deaf people, they may wish to see the transparent video immediately. In this case, there will be a trade-off between loading time and availability. Slow access to the internet could in fact, cause the page to not load at all. The activation icons were kept small in this experiment, nevertheless, they do appear on the page. This could be distracting for the standard user. For those with screen readers, this would also need to be considered. It would need to be obvious what these referred to. If not, they could confuse blind people, as to their use. If web designers were to implement this feature, any upgrade to the site would require careful consideration. The change of content on any page, would require a change to the video associated with that page also. Although it is an extremely good idea, it is not as simple to implement as other forms of accessibility. For that reason, it may be difficult to encourage web designers to use it, despite its huge potential.

Sato et al [14] conducted a study into the current status of web accessibility collaboration between blind users, and volunteer sites and services willing to take part in the project. Several examples were used in this study. Book share, a social scanning book service in the u.s. which provides scanned versions of books for blind people around the world, currently has over 70,000 digital books to over 80,000 users who are blind. The volunteers receive incentives for submitting or validating books, with funding coming from the u.s. Helen2 is a web service, where users can share ratings about other web sites. Utube subtitler3 is a service for utube videos that have captioning. However, there is no way to validate whether the captioning is useful or not. Social labelling is another service, which allows users to share labels for images and landmarks to improve navigation for screen readers should someone wish to use them. Again, there is no way to validate the integrity of these. The study found remarkably, that a lot of users were unaware of their own accessibility problems, and as a result, fewer than expected requests for services were received. The project conducted for this paper allows the authors to improve the accessibility of any web site without the web site authors having to modify their pages. This is done by collecting metadata on a public server. Users and designers of web sites can then come together, and use this data. This process starts with a user submitting a request in the form of a report in their own words, describing the problems

on a particular web site. This is then sent to a volunteer (usually a web designer) who can create external metadata, or provide a direct answer to the original user's questions. The user can then browse the original web site using the metadata, or the explanation given. An example of this is given. A user reports, the problem as "please add headings at http://www.example.com". a volunteer creates metadata with the Uri pattern of "http://www.example.com" and an xpath notation such as "/html[1]/body[1]" with the heading level 2 as the metadata type. The user can now call the function to apply metadata to fix the page using the client-side tool provided. The tool then looks for metadata relating to that Uri. The data is returned if available, and is applied to the page. The user now has an accessible page. The tool is windows resident, and can be accessed via a short-cut key. It is currently supported by Microsoft's internet explorer and Firefox. It can also be used for flash, but the metadata collected here is co-ordinate based, as the flash content may be embedded in the page. Examples of failed and successful requests for assistance were given. Not surprisingly, the failures provided were those requests from users saying that they had encountered maps, and could not understand them. It was found, that even with collaboration between all sides, no one could effectively explain a map. Government sites were found to be easier to fix, as they had static pages. Other sites such as music and video sites were found to be much harder, as they employed some form of flash or dhtml. A further test was conducted before this study was published, as metadata was being collected from 2008. It took all metadata collected up to 01/03/2010, and ran a crawler to see how much of the data was still valid. As expected, some of the pages were unavailable, while others remained static. This was because changes had been made to the pages by the web site owners, but the Uri path had remained the same. 84% of pages were still available. In 41% of the available pages, none of the metadata was valid. 53% of metadata was loaded for visited pages, but only 42% of them were applied. Involving site owners in this process was also considered, but it was not explored in any great depth. While this study is very comprehensive, it does involve creating data to describe data based on the problems of people who may not be aware of what their problems really are. In some cases, "they don't know what they don't know". In the case of flash movies, the existence of such an element may not even be reported. Other elements may also not be reported, depending on the various settings and versions of the screen reading software being used. As can be seen, metadata is not completely reliable. The suggestion was made, that site owners could put tags in their pages, which in turn could access the metadata repository for

their pages. Perhaps, a better approach would be to involve and educate the site owners from the start. Reference was made to the fact, that pages are constantly changing, and new technologies are being introduced. If the problems were fixed initially, then the site owners would be aware of the totality of their accessibility problems. When their content changes, so would their accessibility perspective. There is also the client-side tool that is needed to run this service. When this paper was written, it was only supported by internet explorer and Firefox. Despite Microsoft's investment in html5, its internet explorer is probably the worst platform in relation to support for this version. Add to this a client-side tool to implement metadata based on accessibility problems, an undetermined version of a screen reader, and the problems become too complex to even predict. The study has merit, but perhaps, addressing the basics is the way to proceed.

Kuber et al [15] conducted an evaluation of haptic html mappings in order to improve web accessibility. The study suggests, that blind users of a web site would greatly benefit from being able to touch objects on screen rather than using a screen reader. In fact, the study claims, that some features on web sites were easier to interact with using haptic technologies rather than screen readers. This concept was first introduced with the Optacon, a machine which allowed blind users to read printed material. The machine had a camera, which used a pin array to vibrate the shape of the letter on the blind user's finger, as the camera passed over it. Haptics have been developed for web sites, showing the blind user images and other graphics. However, this technology is recognised as still being very young. The study also points out, that most blind users use a keyboard rather than a mouse. Further training would be required, as the user would not be familiar with the use of a mouse or other such device. It suggests, that a library of haptic mappings could be produced for web designers. It would serve to prevent the same haptic html mappings being used for different representations by developers, and prevent multiple html objects being represented by the same haptic effect, as this could lead to confusion on the part of the user. Multiple design sessions and experiments were carried out to see what the effect would be. A library of haptic effects was developed, and users were trained on how to use them. The study says, that the users were subjected to an unfamiliar web site, in order to gage their responses. However, it is assumed, that this web site was created specifically for the experiment, and therefore, would conform

to the actual haptic html mappings constructed for just such an experiment. The only exception to this could be the search task that users were asked to try. It was noted in the study, that some blind users explored the pages with the mouse very quickly, with the aim of locating a haptic object. After that, users would explore the page cautiously, with the aim of locating another haptic effect. Another user was observed exploring the borders of the page. Once this was mapped, the participant moved closer to the centre of the page in order to locate something. As mentioned above, this was tried out some years ago with the optacon. This machine (it was hoped) would allow blind users to read standard print, and perhaps eventually, have no further use for Braille. However, it proved not to be the case. Standard print was fine, but when letters changed font, or decoration, the process became almost useless. This could also be the case for haptic html mappings. The graphics and text content for a web site can be presented in any amount of ways. In order for such mappings to be effective, every web developer would have to conform to a global standard. Links, buttons and other interactive elements would have to have a defined shape and size for example. Full accessibility cannot be achieved by the use of screen readers, software that has been around for almost two decades at the time of this study. As well as this, the user would only ever have one hand to navigate the web site. Speech software is used for the entire computer, not only the web. If audio queues were played (as is suggested) in order to help the user, this would have a direct conflict with the speech unit, as they would more than likely end up using the same sound card. In fact, the study made reference to speech queues playing when the users were in "white space". Also, the exploration of the screen would be tiresome, as it would take a lot longer to navigate the entire screen, and realise what was in focus. This was mentioned in the study. It is perhaps, a nice idea, but a long way off from being practical. The user is required to locate an element in order to interact with it. This task alone can be frustrating, especially, if the element cannot be seen. With the use of a screen reader, the next element can be located directly by use of the keyboard. In a well designed web site, haptic html mappings would take from rather than enhancing the experience. One of the tasks performed in this study, involved blind users working with their sited counterparts, to fill in a time sheet. Although the tasks were completed, blind users were noted asking for directions when trying to find information on the screen. With time, this would improve. However, with the use of a screen reader, this type of direction would not be necessary. One user mistakenly activated the wrong link, and the sighted user had to direct him to the "back" link. Again, this would

not happen with current screen reading technology.

Brajnik et al [16] conducted a study on the expertise effect on web accessibility. The study conducted a barrier walk-through with 19 expert and 57 non-expert participants. This method is used to do a manual evaluation of web sites for users with different disabilities. With this method, an evaluator uses a list of predefined barriers, a user profile, and possibly a scenario to determine which barriers actually exist and to rate their severity (how they will effect the user profile). The experts in this study were mostly web consultants, while the non-experts were students having attended some sort of web course. Four pages were chosen, representing hand-crafted and professional designed formats. The completion of tasks was done by each participant in their own time, and using their own environment. The study concluded, that expertise did play a significant role when evaluating a web site for accessibility. It was noted, that while three experts could find all problems, fifteen non-experts were required. However, the data did show that experts do evaluate pages differently. There was absolutely no control over the environment, and no mention of how long the study took. There was also no indication as to whether any type of assistive technology was used. The results of this study are very subjective, bringing nothing of consequence to the research, apart from the fact that experts make a difference. Being experts, this was always a given.

Brajnik et al [17] asked, if web accessibility conformance was an elusive property. This paper focused on four web pages that were evaluated by humans rather than machines. The WCAG2 guidelines were used. The set of pages was small, but their configuration (according to the authors) made them a reliable set and produced a wide range of accessibility compliance issues to be investigated. 25 experts and 27 novices were used in the testing. The pages were evaluated online, but care was taken to insure that no major updates took place during the testing. Some small updates did take place, but they were not considered to have any effect on the overall result set. The pages were evaluated online because saving the pages locally would mean that some dynamic content and interaction techniques could not be stored. Before the human study was carried out, the pages were evaluated using an automatic tool to see whether or not

there were errors, and how severe they were. The experts were invited to conduct the study via email, while the novices were students of one of the authors. Participants were allowed to use any evaluation tool, browser and technique that they liked. There was no information collected as to how familiar the participants were with any of the pages tested, however 64% of the experienced participants were accessibility consultants. The paper notes, that the notion of correct grading does not impact on real users, but is based on the findings of the majority of experienced evaluators. They believe that there is no reasonable alternative, stating that other variables would come into play such as versions of assistive software degree of disability and experience of using the various assistive technologies. The experienced evaluators chosen might not correspond to the "knowledgeable evaluator" category that is meant in the definition of reliable human testing. The paper concludes by stating that the expected level of 80% agreement when using human testability for WCAG2 is not attainable. The testing was so open, that nothing conclusive can be gained from it. There was no mention of any assistive technology used, whether or not any of the participants had a disability, and what updates were added to the pages during testing. Furthermore, any version of any system on any computer could be used. There was also no indication as to whether the participants were allowed to use the mouse, or were restricted to the keyboard. In any case, measuring the difference between windows xp and internet explorer 6 compared to windows 7 home edition and internet explorer 10 (to use two examples) would have nothing in common. It is well known, that different versions of software behave very differently when used in accessibility testing. Therefore, the fact that evaluators gave conflicting results when looking at the same page is not at all surprising. Take for example, the current versions of internet explorer and Firefox. If a page is animated in any way, Firefox can deal with it. Internet explorer cannot.

(Marquez et al [18] conducted a study into the new inclusite web accessibility system. This is a system offered in a cloud environment using the software as a service (saas) solution. The idea here, is that web sites will register with the service. Rules are then agreed and stored within the inclusite's server. When a visually impaired user accesses the system, information is returned to them via speech. The system has a standard and expert mode. This can be changed depending on the user's experience. The aim of

the system was to enable users to access web sites from anywhere using any computer. The largest problem that the study found, was that in order to access the site, the participant in the test had to get a sighted user to navigate to the web site, and activate the inclusite software. This had to be done for each different web site. This in itself, almost makes the software unusable. While the idea is good, there are several problems. The largest of these is the one mentioned above. Without someone's help, the software cannot be accessed by a blind user, the ones it is intended for. Other studies mentioned above have also tried something similar with providing metadata and rules for certain web sites. The problems found then still apply now. Once the rules for inclusite have been agreed, the owner of each registered web site will need to notify inclusite if there is any change. If changes are not reflected, it is possible, that the site would become unusable.

1.4 Conclusion

In conclusion, accessibility is an extremely large area, covering a large amount of different and varying disabilities. Although there is legislation and guidelines, they are rarely implemented, and where they are, there still remains inaccessibility. There are a number of different ways to approach this problem. However, with the lack of awareness of developers and management in organisations, it may be some time before any real progress is made. Given the current availability of cloud environments and the ability for people to build their own web sites without needing any training, the problem will only get worse. Add to that, the constant change in technology both from open-source providers, and the providers of screen reader and other solutions, and the problem becomes clear. It may be ultimately more useful, to determine what can be immediately evaluated, rather than trying to evaluate everything. This has been the focus of most of the automated tools currently available.

The aim of this research, is to provide a tool to evaluate web pages, specifically for the needs of blind people. The tool will take an existing web page, produce an error report,

and try to add or change elements on that page to facilitate the screen readers. This page will then be output to be evaluated by human interaction. It is recognised, that not all problems can be fixed by automation, but it is worth exploring. Three sets of data will be analysed, Irish educational sites and 100 random sites from Ireland and Spain selected by a Google search. Once the pages have been put through the software model, the amount of data which has been scanned and changed will be analysed. This should indicate, the level of compliance between the various populations chosen, as well as determining whether there is a significant difference between them or not.

Chapter 2

Background

2.1 Introduction

Web accessibility can be defined as the practice of making a web site accessible to all types of users. The aim of web accessibility is to insure that web sites can be accessed and used by all, independently of the limitations of any individual. This should also include any browser regardless of age, on any type of device (computer, television or hand-held), with any type of connection. This includes users with and without disabilities. Accessibility is something which needs to be considered from the start of a web site's existence. It is not something that can be addressed after the web site has been built and deployed. During development and design, if the site is not properly structured, it can lead to levels of non-accessibility that cannot be addressed at a later stage. It is important when developing a web site to consider things such as layout. For blind users who depend on screen readers, the page will not be presented in the same manner as it would to a user who is looking at it. Blind users generally navigate a web site by use of the keyboard. Until the user becomes familiar with the site, the arrow keys will be used. If the page is not properly laid out, this can lead to confusion about what the various elements on the page are for, and in some cases, what they relate to. In addition, different browsers support different technologies. Web accessibility has been address globally. This can be seen from the various guidelines, legislation and law that has been implemented in different countries. Some countries have also developed their own national guidelines, designed to complement international standards such as those

of the International Organisation for Standardisation (ISO) and the Web Content Accessibility Guidelines (WCAG) versions one and two (http://www.w3.org/TR/WCAG10/, http://www.w3.org/TR/WCAG20/).

2.2 Legislation history

In 1997, the World Wide Web Consortium (W3C) launched the Web Accessibility Initiative (WAI) (http://www.w3.org/WAI/). The main purpose of the WAI was to reinforce the web's basic platform independence, and to provide web developers and designers with specific standards for increasing web site accessibility. Wai was not just concerned with the development and design of web software. It also addressed the user agents such as browsers and media players being used to convey web content. As a result, WAI produced the Web Content Accessibility Guidelines version one (WCAG1) in 1999. These were a series of guidelines developed and presented in document form. They were designed to help developers and designers to create their web sites in such a way as to make them accessible to all. One of the main principles when implementing these guidelines was that web sites should not be made less accessible to one group of users by making them more accessible to another. The emphasis was on making the site accessible and usable for all, regardless of whether they had a disability or not. As technologies evolved, these guidelines to some extent, became obsolete and needed to be updated. This became a reality, when W3C produced WCAG2 in 2008. This new set of guidelines presented several differences to its predecessor. WCAG2 also makes use of the concept of principles that web content should conform to. There were four principles considered to be of vital importance when designing a web site. These were to insure that a web site was perceivable, understandable, operable and robust (POUR).

- Information and interfaces should be perceivable to all users. There should be nothing on the site that is invisible to all of their senses.
- Users should be able to operate and navigate user interfaces without having to receive help to interact with them.
- The users should be able to understand the operation of the user interface, and the information being presented on the web site.

- The content on a site should be robust enough as to be interpreted by a variety of user agents and assistive technologies.

Although there were several differences between WCAG versions one and two, there were still similarities. One of these was the definition of priority and conformance levels. WCAG1 checkpoints were divided into priority levels one, two and three. WCAG2 also adopted this approach. A web site was graded on its implementation of the various checkpoints within these levels. A conformance rating was then awarded depending on the level of compliance. This could be level A, AA or AAA. Priority one checkpoints were those which were considered to be indispensable, if the web site was to be accessible. The second priority referred to checkpoints which would aid in accessibility, but were not absolutely critical for the site. Priority three, referred to features which were desirable, but not important. WCAG2 was designed to be technology neutral. As a result, there was a move away from priorities and more emphasis was placed on conformance levels. The new guidelines were also created in a form that insured they could be evaluated either by automated testing or human interaction. This was to insure, that any web site awarded a conformance level, could and had been properly tested.

2.3 International standards.

Besides these WC3 standards which have been in existence for over a decade now, the issue has also been in the agenda of countries worldwide for a long time. In the USA, concern for disabled people has been in existence since the 1973 rehabilitation act. With the introduction of information related technologies, the need for accessibility became clear to governments. In order to satisfy this need, the USA government adopted the section 508 standard. This forced all government agencies to comply with the WCAG standards, and to make their systems accessible to all, unless this constituted a burden that could not be overcome. Canada has the Common Look and Feel standards requiring government web sites to meet WCAG1 priorities one and two. These standards have been in existence since 2000. In 2008, government web sites were required to follow the WCAG2 standard with compliance to conformance levels one and two.

As part of the web accessibility initiatives in the Philippines the government through the national council for the welfare of disabled persons board approved the forming of an adhoc group of web masters to help with its implementation. The Manila accessible information and communications technologies design and recommendations was drafted and adopted in 2003.

In Spain, UNE 139803 was the national standard used to regulate web accessibility. It conformed to WCAG1. In Sweden, VERVA, the sweetish development administrative development agency was responsible for a set of guidelines used to insure that government web sites were accessible. Through the guidelines, web accessibility is presented as an integral part of the overall development of a site, and not as a separate issue. These guidelines address several areas such as accessibility, usability, web standards, privacy issues, information architecture, content management systems and authoring tools. The latest version of these guidelines was released in 2006. In December 2010, the British Standards Institute released the BS 8878 web accessibility standard. It describes what is required from web sites to comply with the UK disability discrimination act 1995. In Japan, 2004 saw the introduction of the Japanese Industrial Standards X 8341-3 (JIS) for web accessibility. This standard was revised in 2010 to adopt WCAG2. In Ireland, the disability act 2005 required that were a public body communicates in electronic form with one or more persons, the contents of the communication must be, in as far as is practicable, accessible to those with a visual impairment to whom adaptive technology is available. The national disability authority (NDA) produced a code of practice for public bodies, giving guidelines on how to meet the obligations of the act. This was an approved code of practice, and its provisions had the force of law.

2.4 WCAG2 introduction

From WCAG1 (1999) to WCAG2 (2008), one of the main features of the world of ict was its constant technological change. This had the effect of making WCAG1 out of date in a very short time. One of the reasons for this, was that WCAG1 was mainly a technical document. This fact, coupled with the need for facilitating the understanding and use of the guidelines and the right way to test them whether it be automatic or manual, forced the W3C to create a more up-to-date and comprehensive version of WCAG. The development of this was done with collaboration taking advantage of what

had been said about WCAG1 by internet users and experts. Another aspect of this new set of guidelines was the production of manuals and hand-books to aid developers and designers with the implementation of the new standard. The documents also helped to inform developers how to switch easily and efficiently from WCAG1 compliance to the new WCAG2 standard. There were however criticisms about WCAG2 by some web designers and those directly involved with its production. In an article published in May 2006 by Joe Clarke (http://alistapart.com/article/tohellwithwcag2), various problems were pointed out in relation to WCAG2. He considered, that the documents produced to explain this standard were impossible to understand. There were also problems with the working group established to produce WCAG2. Only 34 days were allowed for comments after the drafts were produced. This included the entire industry, as well as people with disabilities and their organisations. Some of the criticisms included:

- What constitutes a page or site is a matter of dispute.
- A web site that complies with WCAG2 may not have valid html encoding.
- Multiple nested tables can be used for page layout.
- Parts of the page can blink for up to three seconds, however, the page may not flash.
- Entire sets of technologies can be defined as a base line, meaning that anyone without this technology will not be able to access the site.
- Entire directories can be defined as off-limits to accessibility. In WCAG2's own example, this can include freestanding videos.
- If videos are posted online, an audio description is no longer required for blind people.

Several hundred navigation links may be placed on a single page. However, if the site has two or more pages with navigation links, a "skip navigation" link must be provided. An alternative page must be produced for those who cannot understand the main page. The definition given is for those with a lower secondary education level who are trying to access the site. One of the most important consequences of these criticisms is that developers and designers are still using WCAG1 as a standard to make

content available and accessible on the web. In 2013, W3C published the WAI-ARIA (advanced rich internet applications) guidelines. These were designed to help developers and designers when producing web sites that contain AJAX (advanced JavaScript and XML), JavaScript, Html5 and the use of new roles such as Menu Item. The use of these technologies have an effect on the way the page is presented and changed. It is entirely possible, that JavaScript can be used to add, remove or change elements on a page after it has been loaded. There is also differences in the way that these new technologies are supported across browsers. In 2013, support for html5 by Microsoft's Internet Explorer was practically non-existent. As a result, it was difficult to use this browser with a screen reader, as many of the items either did not display correctly, or did not display at all. Consequently, the blind user did not get a clear picture of what was contained on the page.

2.5 Accessibility and what can be tested

Generally, the accessibility of any web site relates to seven separate components (http://www.w3.org/WA
However, the lack of any one of these can have a serious effect on the site's accessibility. These are: The web site itself, including the information (text, images, sound) and the structure of the mark-up code used to define it. User agents, (browsers and media players). Assistive technologies (screen readers, magnification software and hardware devices). Developers, the methods tools and knowledge used when developing the web site. Authoring tools (Dreamweaver, Microsoft's Visual Studio, and frameworks such as JavaScript and JQuery). Evaluation tools, such as SortSite. A defined web accessibility standard (including but not limited to an internal standard for an organisation). In order to test a web site for accessibility, there must be a human element in the testing as well as automation. These two separate parts of the test can be referred to as the "judgement" and "algorithmic" parts. There is no single tool, that when run against a web site or page, can say with absolute certainty that the site or page is compliant with the WCAG standard or any national standards, regardless of how good or expensive it is. When a web site claims accessibility in relation to these standards, the first question that must be asked is how much of the testing was "human". Many sites claim accessibility having used tools to evaluate their software. The best that can be claimed from this, is that the site or page passed all the automatic tests provided and run by whatever tool was

used. Tools do play a vital role in accessibility testing. They can be good indicators as to whether the site has been developed correctly in the first place. A complete list of current tools from the W3C site is available at http://www.w3.org/WAI/RC/tools/complete. What accessibility testing is possible? To answer this, the following section explores the elements that can be tested and whether it can be done by automatic tools (algorithmic) or whether it requires human intervention (judgement).

2.5.1 Text equivalents

Algorithmic. Software can determine the presence of alt attributes on elements where they are required such as img, area and input with type = image. A check can be done to see exactly what is in the alt string. This in some cases can be null (alt = ""), but is never desirable on an element that can receive the focus at any time while interacting with the site. Common errors and violations can be detected also such as the use of file names in the alt attribute. The "space" character can also be checked. Judgement. Judgement should be used when evaluating the content of the alt tag. Does the text relate at all times to what an image or button may be doing? If an image becomes inactive is this reflected? Alternative text for audio files would have to be checked by human interaction. If the recording is that of a speech for example, it would have to be verified, word by word. Tools can use metadata to determine whether the text is or is not valid, but there is still a margin for error. Synchronised multi media. Algorithmic. The existence of multimedia on a page can be checked. This can be a file extension in an anchor, or the content of an object element. However, it may be difficult to check whether captioning is present. Perhaps the best way to handle this, is to highlight the existence of multimedia in a report. Judgement. Human interaction will be required initially, to insure that captions are present where required. Then there is also the question of whether or not the captioning is valid. If an audio description is required, is this valid and usable? Colour coding. Algorithmic. All information conveyed with colour should also be available without. Colour specifications could be checked in the style sheet, and against the corresponding elements on the web page itself. If there is no colour change, then the site can be considered to be compliant as far as text is concerned. However, images on the page such as link images could still be colour coded. Other images could be displaying important information. If there were colour changes within the page or style sheet, it would be difficult to determine whether this

was having an effect on the information being conveyed. A search of the code for colour words could produce irrelevant errors. Take for example a page with the sentence "the books in green are reduced in price today". Algorithmic checking would produce an error when reading this, although it is totally irrelevant. Judgement. A general evaluation is necessary to determine what effect colour is having on the display of information. There are some who suggest, that the best way to do this is to view the page in black and white. However, it would be much more effective to view the page in its natural state. After all, that is the way the entire world will see it. Viewing the page in black and white, alters the testing conditions, and can present its own problems. Style sheets. Algorithmic. Software can check for the presence of style sheets or their elements. In order to conform to the web standards, documents should be organised in such a way as to make them readable without the need for a style sheet. If no style sheet elements are detected, then this makes the document compliant. However, this is highly unlikely. Most web developers use style sheet elements in order to decorate their sites. This has been common practice almost since the internet became available to the general user. Beyond this, human interaction is required to observe how the site looks and operates without style sheets. This should be done, as screen reading software operates on the basis that the site has no style sheet. Judgement. Pages should be viewed in a browser with style sheets turned off. It is also a judgement call as to whether structural elements are being presented on a page using style sheets, or html elements such as headings and footers. Maps. Algorithmic. If the map is server side, redundant links should be provided for each active region on the map. Client side maps should be provided wherever possible. The exception to this is where the active regions cannot be defined with an available geometric shape. A program could be written to click on every pixel of a server map, record the new url, and compare it against every link element on the page. However, this may be overkill. These type of maps could be flagged for human intervention. Since all regions of a map can be defined with polygons, the page should fail if the map is server side. All areas of the client side map should have a valid alt attribute. Judgement. If the map is server side, it must be evaluated for text links for all active regions. If the map is client side, then the page is compliant. Regardless of the map being used, it is irrelevant to screen readers, since the map itself is an image, and cannot be read. However, some sort of text should be provided indicating that a map is present. This cannot be evaluated by software. It is also possible, that a map may become part of a page at a later stage. This can be done by the use of JavaScript. This

can only be observed by human judgement. Table headers. Algorithmic. If the page has no tables, then this is not an issue. However, tables are still extensively used. They can be data or layout tables. There are certain checks that can be carried out in order to evaluate what a table is. It is possible to devise algorithms to separate out these tables, but they are extremely sophisticated, and may not always be reliable. The use of images in cells would suggest a layout table, whereas uniformity of rows would suggest a data table and so on. Nested tables are almost certainly not data tables, however this cannot be automatically assumed. If a data table is determined, the software could check to insure that proper mark-up is being used where it should. Row one column one should be a "th" for example. If "th" elements exist in rows one and two of a table, then there is more than likely, a problem. Judgement. No matter how sophisticated the software is, there is the possibility, that tables will be incorrectly classified. Human interaction is required to insure that the proper mark-up is being used in its proper location, especially in data tables. Complex tables. Algorithmic. If a complex table is found, it is practically impossible to determine whether or not it is compliant. These types of tables can be classed as both layout and data. Judgement. Human interaction is required in this case to insure that correct mark-up is being used to display the relevant elements in an accessible fashion. Frames. Algorithmic. Software can check to see if frames in a frame set have relevant title attributes. Errors should be placed in the report where the software is unsure. Frames are widely used in web sites. These can include Google adds, IFrames and general html frames. Judgement. Human judgement will determine whether or not the title attribute is useful. In a large amount of web sites, frames are titled with a number (e.g. 1077) which indicates nothing. Two of the main interaction frames that are known for this are FaceBook and Twitter. If a frame has no title, a screen reader will use either the title of the web site being displayed within the frame, or it will read out the link pertaining to that page or site. Regardless of whether the title is to be used or not, it should be present. Flicker rate. Algorithmic. Software can determine whether or not technologies are being used on a page that cause flicker problems. These can include but are not limited to Flash, Java and JavaScript. Furthermore, any animated gif on the page can also be responsible. An elevated flicker rate does not necessarily mean that the screen is flickering at that rate. Screen flickering is measured as the change between light and dark. Moving text can also be detected. Judgement. Only human interaction can determine whether or not the page complies. However, if flicker technology is detected, it should be flagged for inspection by the

testing software. This can have a serious effect on screen readers. If images are being resized, this will effect the location of the cursor on the page, causing the screen reader to change location without the knowledge of the blind user. Text-only pages. Algorithmic. The standards say that text only pages should be provided where compliance cannot be achieved in any other way. The page must also be updated at the same time and with the same frequency as the primary page. Although it is possible to detect text only pages with software, it is not possible to insure that they are being updated with the same frequency as the primary page. In fact, to determine this by software could be misleading. If the primary page is simply being updated by an animation that has no relevance to the information, this would give rise to incorrect information being reported by the software. Judgement. Firstly, is it really not possible to make the primary page accessible in any other way? Secondly, having determined that a text page is required, is it accessible? Is one available? Does it offer the same functionality as the primary page? Does it comply with all other accessibility standards? Is it kept up to date side by side with the primary? This is probably one of the most difficult tests to get right, as the human factor is quite extensive. Scripting. Algorithmic. If scripting is used to display content or to create elements, this content should also have relevant alternative text that can be read by screen readers. For a piece of software, this task is extremely complex. It is in effect, one program trying to evaluate another. It is possible, to save multiple copies of the page as it is modified, and run tests on it. However, in a very complex page, this could become extremely time consuming. A single change to the script could make any test carried out before, completely invalid. If the attributes are modified for a visual effect, it may not be as important. However, it could have an effect on those using magnification programs. Judgement. The easiest way to evaluate this is to have someone use the page with assistive technology. Fly-over menus can be tested to insure that they have valid links when required. If these alternative links are not available, then the modification is visual, and must be addressed. A prime example of this, is the BlogSpot web site. Users of screen reading software cannot access the fly-over menus, as they are completely visual. Their appearance also changes depending on the browser being used. Applets and plug-ins. Algorithmic. In some cases, the user is required to install an applet or plug-in in order to run the web page requested. The page should provide a link in order that the relevant plug-in can be obtained. This applet or plug-in should also in itself be accessible. It is possible to check to insure that the link on the page is present. However, it is not possible to check the download requested.

Judgement. Human interaction is required to insure that the applet or plug-in being used can be installed by use of the link, and that afterwards, the page or site does not become inaccessible as a result. It is also possible, that some further testing may be required after the installation. Sites that require the presence of a third-party piece of software must be usable and accessible both before and after the software has been installed. Online forms. Algorithmic. Software can check that every input element has a corresponding label, or relevant alt tag. The same must be done for other controls such as combo boxes and menu items if present. Once the form has been filled in, there should be some change to the page to indicate this. It may not be possible to check this with software, as the submit button may run a script function. Judgement. Human interaction is required to verify, that the prompts and actions in a form make sense. Software can evaluate that labels and titles exist, but as with alt attributes, they may not be accurate. As already mentioned, it should be verified, that something happens to the page after the form information has been submitted. In some web sites for example, the form is simply blanked. This can make the user believe, that their information was never submitted. However, if the form changes, and a "thank you" message is displayed, the user will be in little doubt as to what has just happened. Skip navigation. Algorithmic. This is not an exact science. Software can check to see if a link on a page is attached to another local link. However, this does not necessarily indicate that it is a navigation skip link. It may simply skip over its current list. Frequent use of heading levels can also be checked for. This in some cases can be just as effective as navigation skip links. Tables and frames can also be used for this, but there is no way for software to determine that that is their specific use. Judgement. Human interaction can examine the page to see if there are large blocks of links. If so, is there a "skip" link present in the list? If not should there be? Time responses. Algorithmic. If there is no meta-refresh, there are no forms, or if forms are present and there is no script, then there is no time responses. Judgement. Check to see if the user can request more time. If not, is the time allowed sufficient? If the site is being used by someone with a screen reader, they may require more time to locate input fields, or read relevant information before the site is refreshed. Refreshing a site too soon can have the effect of resetting input fields to blank. This can greatly frustrate a blind user, causing them to abandon the use of the web site. Duplicate anchor tags. Algorithmic. It is possible for software to check the presence of duplicate links, or links with irrelevant text such as "click here". Very often, developers will have parts of lists on the site with a "more" link at the bottom of

the list, indicating that the list can be expanded. This can be extremely frustrating for those using screen readers. Jaws for example, allows the user to access all the links on a page by use of a shortcut key stroke. The list is presented in alphabetical order. This has the effect of placing all the "more" links together, without giving any information as to what they belong to. Judgement. It is necessary, for someone to investigate what these various links are for, and to whether or not, they need to be changed. A "click here" link for example, could reflect what needs to be clicked. The "more" links could reflect what they will provide more access to when clicked. Image tags. Algorithmic. These tags can be checked by software to ascertain whether they have alt text, width and height attributes. Again, as with link tags, there can be duplicates, and meaningless text. If there is no height and width attributes, this should be flagged by the software. Judgement. It is valuable to have these types of errors checked manually. The alternative text can give meaningless descriptions as to what the image reflects. If the image has no width or height attributes, this will cause screen readers to lose their place on the page as it is loaded or refreshed. Because there is no predefined area on the page for the image, it will move around as the page is manipulated.

2.6 Conclusion

Almost everything that can be evaluated, can have both an automated and human set of procedures. There is no doubt, that software evaluation tools make an invaluable contribution to the web accessibility testing process. With that in mind, this research has focused on developing a software tool which will specifically evaluate web pages for blind users. There are many different types of disabilities, all requiring different features in order to make web sites accessible. For the purposes of this research, the tool was designed to test pages for those elements which effect blind users using screen readers specifically. The Web Assessment Accessibility Model (WAAM) software tool, will produce a report for each web page, and will attempt to produce a modified version which will be accessible to blind users. It is noted, that regardless of how accessible the modified page is, it should be evaluated by human interaction. There are a great many evaluation tools available for web sites. These tools provide the user with reports concerning various problems with the site and its pages. Some of them attempt to validate html code. Until now, the focus has been on providing tools that try to capture

all accessibility issues. In reality, this is very difficult, as every disabled category of user has their own needs, and in some cases, they can be contradictory. A user with dyslexia, may prefer to have the page read to them by use of an audio mp3 file. However, for those using screen readers, the speech from the reader may be over-powered by the audio file. It is hoped, that by producing a modified page as well as a report, that WAAM can provide web developers and designers with a greater awareness of what is required. They will be able to see the changes that have been made, and as a result, they should be able to improve the overall accessibility of their web sites.

Chapter 3

System Model

3.1 Introduction

Accessibility has become a major issue with regards to web site design. There have been a number of attempts to address this, from international guidelines and standards to local and national legislation and laws enacted in various countries throughout the world. However, a lot of these standards and guidelines are focused on insuring that government sites conform. There is very little to enforce accessibility on web sites that are not government related. Developers and designers of these sites include accessibility, only if it is practical, and even then, accessibility still remains poor. There is also the problem of the various tools being used in the development process. Adobe's Flash allows the developer to place embedded flash frames on pages with unlabelled controls. An unlabelled button would be announced by screen readers as "unlabelled 3 button" for example. Microsoft's Visual Studio 2012, produces web sites with accessibility issues by default. A standard web project generated from this tool, has serious flaws, e.g. each heading level appears as the one below it. A heading level one, will be announced by screen readers as "heading level 2". The JQuery user interface framework, has controls available in it such as the date picker, that cannot be accessed. Automatically, as web sites are developed using these tools, they develop accessibility issues, and in some cases, become inaccessible completely. Cloud providers and those companies that allow users to develop their own web sites, also contribute to the problem. Even these basic sites can be inaccessible from the start. As a cloud provider, Engine yard cannot be used by blind developers, as its main page is constantly changing, confusing the location of

the cursor for screen readers. For the purposes of this research, A query was opened with Microsoft in January 2013, requesting that they look into the fact that the Visual Studio 2012 help download window could not be accessed by the Jaws screen reader. This allows the user to download help files directly on to their local machine. In July of that year, the query had still not been addressed with any grate efficiency. Given these facts, accessibility will continue to suffer, especially for the blind user.

3.2 Software model

The Web Assessment Accessibility Model (WAAM) software, is designed to address this problem. It is capable of analysing a single page, or an entire site for certain criteria. The software will output an accessible version of the pages, and reports for each page scanned indicating what has been changed, and what requires human intervention. For example, images with no height and width cause screen readers to lose focus as the page changes or loads. The software has no way to evaluate this, so it is flagged in the reports. It is hoped, that web developers, designers and maintainers will use this model to make their sites more accessible. There are many categories of disability, each requiring their own levels of accessibility. For the purposes of this research, the focus is on making sites accessible for blind users specifically.

WAAM was developed using Microsoft's Visual Studio 2012. Despite its short-comings where accessibility is concerned, it was the most appropriate development environment for this project. The operating system used was Windows 7 home edition. All updates were present at the time of development. It was designed to run off-line. There are a lot of other tools on the market, which can evaluate pages online. This can lead to problems. The page or site can be subject to constant change. A test run now, may not yield the same results as a test run one hour later. There is no way to be sure whether or not the page has been changed in that time. For that reason, this software has been designed to work with off-line pages and sites. Each page to be scanned has to be downloaded, and placed on the user's machine.

3.2.1 How to use the software

The tool can be run in two different ways, and has two different interfaces. There is a web interface, which allows validation of a single page at a time. The user will be asked to provide the location of the downloaded page file. This file will then be scanned, and a report will be presented on screen. This report is also saved in the same directory as the scanned page. A modified accessible version of the scanned page is also available in this directory. The second interface allows the user to run the tool from the command-line. It asks the user for the directory in which the page or pages are stored. This part of the software is capable of scanning a number of pages at any one time. It is windows based, as a connection to the web may not be available, or the amount of pages needing to be scanned could be large. This will produce a report for each page scanned, and a modified version of that page. A site report is produced showing overall totals of what was scanned and what was changed. From the output of either interface, the modified pages can be re-scanned. However, the software will not fix all the problems noted. Reports will tell the user that their pages contain JavaScript, or that they are advanced rich internet application (aria) pages. These messages are only informational. There is no way to evaluate what effect aria roles or JavaScript functions will have on pages when they are rendered in a browser. This is mentioned in the reports, as human interaction would be required in order to see if either aria roles or JavaScript has an effect on the accessibility of the relevant pages. The software functionality is further explained below.

3.2.2 Web Version

The user is first presented with the home page, containing the validate page link. Once this link is clicked, a screen is presented, asking for the name of the file to be uploaded and evaluated. The user can browse for a file location, and select whatever file they choose. There is no restriction on file names, as it is possible that the web page may have been saved with a different extension. Once the file has been located, it appears in the file name box on the page. The user can then either validate the page, or return to the home page. If the validate page option is chosen, a modified accessible version of the page is produced along with a report. The elements that are scanned, and the potential changes are explained below. Once the validation is complete, the report is presented on screen. This report shows everything that has been done to the page during validation.

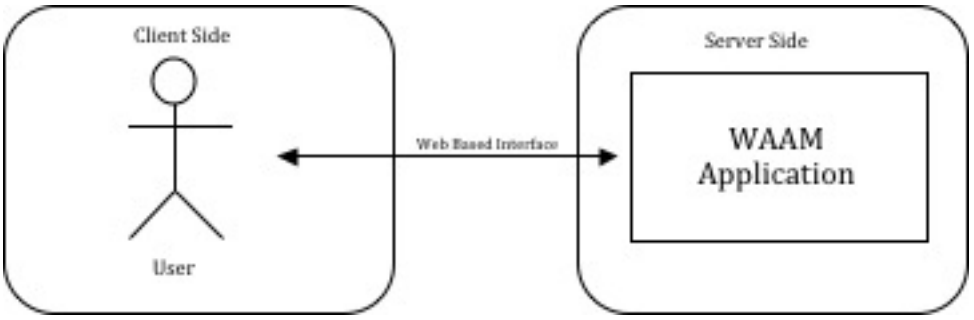

FIGURE 3.1: High Level architecture for Web Based Model interaction

If elements have been changed, the original html line is shown, followed by the changed line. At the end of the report, statistical messages and warnings (if any) are displayed. The option of returning to the home page is provided at the bottom of the screen. These reports are saved on disk for further reference. After looking at the reports, the user can modify the original page, and have it scanned again. Obviously, the results will change. This web version has been tested with Microsoft's Internet Explorer version nine, Mozilla's Firefox version 21, and the Jaws screen reader version 14 from Freedom Scientific.

Figure 3.1 gives a high level overview of the web architecture for the WAAM application. On the left is the Client side with the standard stick-man illustrating the user, on the right is the Server side, this box contains a smaller box which represents the WAAM Application. There is an horizontal line with arrows at either end titled 'Web Based Interface', connecting both boxes, this represents the fact that there is two way communication between the user, using a web browser, and the application via http connection.

Figure 3.2 shows the Class Diagram for Web Based Application, illustrating the user interacting with the HomeController class which in turn calls the PageValidator class. The diagram also illustrates the User's interaction with the SiteDownloader class which was only used in the research phase of the thesis.

3.2.3 Windows Version

This is a command-line stand-alone version of the WAAM software. It is designed to process a single page, or large volumes of pages at any time. It requires no interaction from the user, other than when it is first activated. The processing will continue in the

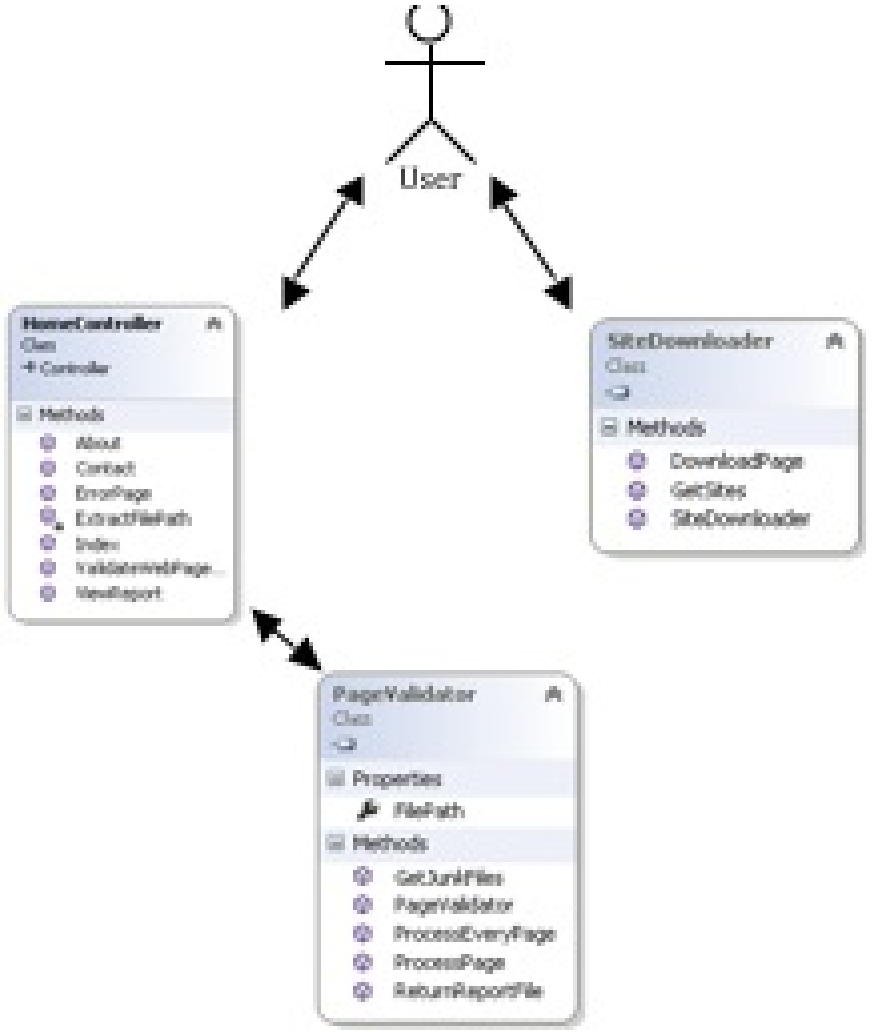

FIGURE 3.2: Class Diagram used for web based application

background for each page. Once the process is finished, a message will be displayed indicating that the modified pages and reports are ready. Once activated, the user will be asked to enter the path to the directory where the web pages are stored. The software will attempt to locate the directory. If it is not found, a message will be displayed, and the process will terminate. Once the directory has been located, the software will search for any ".htm" files. It is assumed, that saved pages will have this extension. Unlike the web version, the windows version scans the directory, and processes all files within it. This will happen for one or many files. It is therefore assumed, that the files will end in ".htm". it is possible, that other files may exist in the same directory. To process these, could lead to errors and misleading information in the final reports. As with the web version, the software produces a modified page and report for each page processed. It also produces an overall statistical report, giving counts of what was processed, scanned, changed and warnings (if any). Unlike the web version, these reports are not displayed.

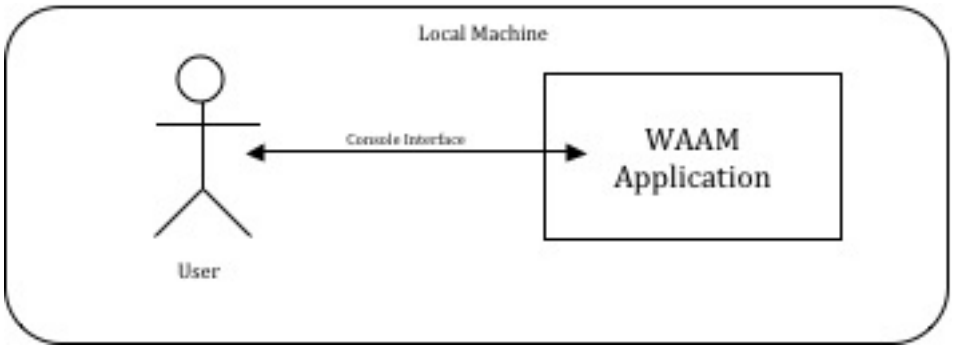

FIGURE 3.3: High Level architecture for desktop based application

They are saved in the same directory as the original page files for inspection. Once the scan has been run and the reports evaluated, the original pages can be modified and the scan can be run again. Obviously, if pages are changed, the results will change also. This software has been tested using windows 7 home edition with all updates as of July 2013. There is no version of this software for any other operating system.

Figure 3.3 gives a high level overview of the architecture for the desktop WAAM application, displaying one large box titled 'Local Machine'. The stick-man representing the user is placed inside the box to the left and the smaller box representing the WAAM Application is placed inside the box to the right. There is an horizontal line with arrows at either end titled 'Console Interface', connecting the user with the application, representing the fact that the user interacts with the application locally.

Figure 3.4 shows the Class Diagram for Console Based Application, illustrating the user interacting with the Program class which in turn calls the PageValidator class. The diagram also illustrates the User's interaction with the SiteDownloader class which was only used in the research phase of the thesis.

3.2.4 Technical description

Regardless of the version used, each page is processed in exactly the same way. The file is selected, and passed to the validation module. In the web version, this is done via the validate web page. Only one file can be passed. In the windows version, each file in the directory listing is passed one at a time. A directory listing is obtained, after the user inputs the path (windows version). All files from previous test runs are overwritten. There are four parameter files required to run this software. If any one of these does not

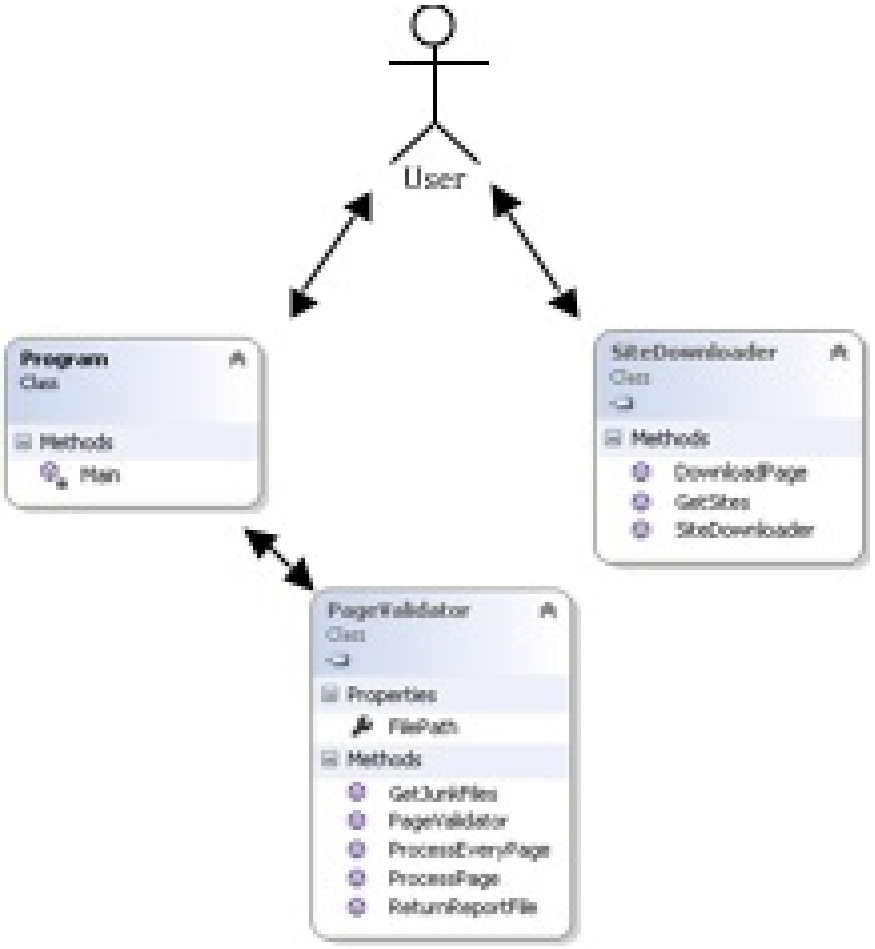

FIGURE 3.4: Class Diagram used for desktop based application

exist, an error message is displayed to the user, and the process terminates. These files are ImgJunk.txt, IFrameJunk.txt, HRefJunk.txt and TagsToBeIgnored.txt. the first three files contain information that is not desirable when processing images, IFrames or links. An example of this would be "click here" as link text. This to the blind user means nothing. The fourth file contains tags that should be ignored if found. Some web pages contain tags that are just opened and then closed. This can be confusing when validating the file, and are ignored. These files should be resident in the same directory as the page or pages to be evaluated. They are parameter files that can be created by the user. As a result, any page or set of pages can be scanned for different values, depending on requirements. The software will not work without them. At start-up, each file is read into memory. Each entry has all "white space" removed, and it is converted to lower case. There is no way to be sure, how information has been entered into these files. For this reason, they are converted into a usable format after being read. In every case, "click here" is automatically scanned for, as it is quite common in web pages. The

page file is opened and read into an array. This allows it to be scanned one line at a time. The file is also converted into a single string. The full file string is checked to see if it contains the aria menu item role. If it does, it is classed as an aria page, and this will be shown in the final report. There is no way to conclusively evaluate aria pages. For screen readers, it is important to have the various roles properly inserted into the html source. There is also the problem of browser support. The presence of JavaScript is also checked. If it is found, it is shown in the report. Because JavaScript cannot be analysed, it is impossible from the source code, to tell whether this will make the page inaccessible or not when it is activated. The web page may only make reference to JavaScript files. The string is then check to see if it has any heading levels. These range from one to six. If not, this is also flagged in the report. However, a heading level one is inserted after the body element. If there are link tags in the file with no link text, the link receives a "click here" text label. These cannot be ignore, as they may be place holders for JavaScript processing. This is only required if the link does not have an image attached. If it does, the images are processed separately. Once these checks are complete, the file array is processed. Each line in the array is checked for image, anchor and IFrame tags. As of July 2013, only these three tags are scanned for.

3.2.4.1 Images

Image tags can be included in web pages in a number of ways. They can be sources for link backgrounds, placed on the page via JavaScript, or placed on the page just to convey a picture. Regardless of what they are used for, they will have a "¡img" tag at the start of their declaration. There are various attributes that can be specified such as alternative text, height and width. There is no procedural syntax for where these can be specified. This depends on the author of the web page. In fact, an image can be placed on a page with no attributes at all other than its source. The selected line is split into image tags, as there could be more than one image specified. Each line is now processed. It is converted into lower case, All white space is removed, and any single quotes are replaced with double quotes. This is necessary, in order to determine whether or not the various attributes are present. The "alt" attribute could be entered as upper case, with spaces or tabs between it and its "=" sign. Single quotes can be used in html just as double quotes can. By removing white space, and converting to lower case, the "alt" attribute, if it is present, will read as "alt=". The line is checked to see if height

and width attributes have been specified. If not, this is noted for the report. Height and width attributes cannot be added, as there is no way to determine the dimensions of the image source. This has a serious effect on screen readers when the page is being changed or loaded. The line is checked to see if the alternative text attribute is present. If not, it is added with text which must be reviewed by the user. Screen readers cannot operate successfully without this text. The image is either ignored, or its file name is spoken. If the alternative text attribute is present, the text relating to it is processed. It is almost impossible to validate this text, but certain phrases can be looked for such as "click here" and "banner". These are quite common. The phrases to be checked for are held in a file, which is read by the software at start-up. If one of these phrases is found, it is replaced with text. This text should be reviewed by the user.

3.2.4.2 IFrame tags

IFrames are included in web pages and can show different types of content. They can refer to an external page, or they can have their own content. If the IFrame refers to an external page, this will not be scanned, as it will not have been downloaded. IFrame tags can have a number of attributes. The attribute checked by the software is the "title" parameter. Each IFrame should have a title. If it does not, a screen reader will either speak the title of the embedded content (if there is one), or it will speak the IFrame tag itself. The selected line is split into IFrame tags, as there could be more than one IFrame specified. Each line is now processed. It is converted to lower case text, all white space is removed and all single quotes are replaced with double quotes. This is necessary, in order to determine whether or not the title attribute is present. The "title" attribute could be entered as upper case, with spaces or tabs between it and its "=" sign. Single quotes can be used in html with the same effect as double quotes. By removing white space, and converting to lower case, the "title" attribute, if it is present, will read as "title=". The line is checked to see if the title attribute is present. If not, it is added with text which must be reviewed by the user. If the title attribute is present, the text relating to it is processed. It is almost impossible to validate this text, but certain phrases can be looked for such as "title" and "frame 1". These are quite common. The phrases to be checked for are held in a file, which is read by the software at start-up. If one of these phrases is found, it is replaced with text. This text should be reviewed by the user.

3.2.4.3 Anchor tags

Anchor tags are used in html to form links to other web content. This can be another page, another link on the same page, or a JavaScript event which performs some processing and can in some cases, change the page content. These must be clearly constructed for screen readers. An anchor can have a number of different attributes. They can also have other tags embedded in them such as the span or image. Anchors can also be used as a place holder. In this case, the tag is defined, but nothing is placed in it. This can be confusing for screen readers, as it is not properly identified. By the use of JavaScript, this can become active, but the screen reader will not be aware of it. The analysis for this tag is done by making reference to the text which identifies it as a link. This is usually located just before the end of the tag, but this is not always the case. Those that have an image embedded in them are ignored. This is because images have already been scanned. It is possible for the image alternative text to act as the link text. The line is split into anchor tags, as there could be more than one link specified. Each link line is now processed. It is converted to lower case, all white space is removed and all single quotes are replaced with double quotes. This is necessary, in order to scan the link text for various phrases, and to ascertain whether or not the href attribute is present. Attributes can be entered as upper case, with spaces or tabs in them. Single quotes can be used in html with the same effect as double quotes. The line is checked to see if the href attribute is present. If not, it is added with text which must be reviewed by the user. A similar check is done for blank links e.g. "¡a¿¡/a¿". It is important for anchors to have a href attribute. If they don't, screen readers will not identify them as clickable links. If the href attribute is present, the text relating to it is processed. It is almost impossible to validate this text, but certain phrases can be looked for such as "more" and "click here". These are quite common. The phrases to be checked for are held in a file, which is read by the software at start-up. If one of these phrases is found, it is replaced with text. This text should be reviewed by the user. Duplicates are also checked for. As each link is processed, it is added to a list. Each link is compared with those on the list. If a match is found, a number is added to the end of the link text. E.g. a link showing "more" could be at the end of a list. There could be several lists, each with a "more" link. This is confusing for screen readers. As duplicates are encountered, the count is added to the link. Therefore, the first link will be "more" and the second

occurrence will be "more1". This should be reviewed by the user. This is also recorded and displayed in the page report.

3.2.4.4 Modified page

As each item above is scanned, there may be changes depending on what has been found. Regardless of whether there are changes or not, a modified version of the page will be output from the process. If there are no changes, this page will be the same as the original. Otherwise, the modified page will reflect the changes that were made during the validation process. Each of these changes should be reviewed by the user. The replacement text is not meant to be compliant with the page's structure. It is simply modified, to reflect that there is a problem with accessibility. E.g. if there is no heading structure, a heading level one will be inserted after the body element. This heading may have text like "main content". Obviously, this is not reflective of what the content is. However, it does highlight the fact that it is required in order to make the page more accessible. The modified version will have the same name as the original page, with a modified extension. If the original page was "page.htm" the new modified page will be "page.modified.htm".

3.2.5 Reports

3.2.5.1 Individual page report

Each file processed will have an individual report attached to it. The name of the report will be the file name followed by ".txt", e.g. "page.htm.txt". this report will show each line of the original page that has been changed, and its new structure. This should be reviewed. At the end of this report, various statistics will be presented. These include a count for number of images, links and IFrames that have been scanned and changed, number of duplicate links and images with no height or width, and Percentages for each of these. There can also be a number of warning messages present. If no heading structure was detected, the report highlights this, and the fact that a heading level one was inserted into the document. The report also shows if aria roles, JavaScript or images without height and width attributes were detected.

3.2.5.2 Overall site report

If the software has been used to process more than one web page, a site report is also produced. Unlike the page report, the site report contains statistics relating to the entire run. The numbers are similar to the page report, but there are no warning messages included. Where JavaScript and other warnings are detected, they are counted. This report provides information on how many pages were found with potential problems, what they were, and what was fixed by the software. The individual reports should be reviewed for the details of any changes. The site report is designed to provide statistics such as those mentioned above. It also includes averages, percentages, minimum and maximum values for each item that has been changed.

3.3 Conclusion

There are many tools on the market, that are capable of producing such reports. These vary in what they evaluate. However, there is no piece of software available to produce modified pages for blind users. Because the page has to be reconstructed with the various changes, there are limits on what can be evaluated. When scanning a link, there can be a number of other tags embedded within it. An anchor can have a number of span tags within it, each performing their own task. Each of these can have text displayed, which makes up the link text. In such cases, only the last piece of text is evaluated. Because WAAM is designed to provide accessible pages for blind users, style sheets are not considered a problem. These are mostly used to determine a page's colour, text size and other visual attributes. As these make no difference to a screen reader, they are not evaluated. There are other tags that can be checked. However, this initial prototype, is designed to show that modified pages can be produced, that doing so makes a difference to the user, and to provide proof of concept.

Chapter 4

Research and Analysis

4.1 Introduction

The aim of this research, was to provide a model to evaluate web pages, specifically for the needs of blind people. The model takes existing web pages, produces error reports, and tries to add or change elements on these pages to facilitate screen readers. These modified pages are then output to be evaluated by human interaction. It was recognised, that not all problems can be fixed by automation. Accessibility of web sites is a major problem facing disabled people today. It is something which is often forgotten about, or web developers and designers believe, that it will have a major cost impact to implement. This is not always correct. However, it does depend on the development strategy used. If implemented properly, it can enhance the business potential and usability of a site. It can be the difference between a user returning to the site over and over, or leaving the site because of its lack of usability.

4.2 Methodology

This research focused on accessibility for blind users specifically. The research asked, can web sites be evaluated and modified successfully for blind users using a software model? To answer this, a quantitative methodology was chosen. Qualitative methods could have been used, but these would require testing of the chosen web sites specifically by blind users in a very tightly controlled environment. This environment would have

been difficult to produce. Blind users use a variety of different technologies to access web content. Each of these technologies vary in what they can and cannot recognise. Physically, the access is also different. Users of Braille displays will access web content differently than those using screen reading software. There are also different versions of browsers, operating systems and output hardware. As a result, there would be problems in finding a large enough group of blind users to conduct testing. Not every blind user is familiar with the Jaws screen reader package. As a result, asking them to test a web site using this product would prove problematic regardless of their overall i.t. experience. Because of this, a quantitative method was chosen.

4.2.1 Model

A piece of software was produced, which was capable of analysing web pages, producing a modified version and a page report detailing all changes that were made, and warning messages for those changes detected, but could not be addressed. This is explained further in the software contribution section of this document. The Web Assessment Accessibility Model (WAAM) approach, collected data on evaluated pages, showing what was found, what was changed and what would need direct human intervention. The measure for success was the amount of items changed, versus those identified as potential problems. The more changes that can be made by the tool, the more accessible the modified page will become.

4.2.2 External Factors

Because a quantitative method was chosen, the environment was carefully monitored. WAAM was designed to run with web page sources. There was no online interaction with web pages. This insured, that pages being evaluated did not change between experiments. There are various tools available for online evaluation, but this approach was considered to be unusable, as there was no way to tell whether the page had been changed between tests. Each site to be scanned was downloaded. There are various forms of assistive technology that can be used by blind users. These experiments were conducted using the Jaws screen reading software version 14. Microsoft's Internet explorer version 9 and Mozilla's Firefox version 21 were used as browsers. Microsoft's windows 7 was used to run the software, and conduct the tests.

4.2.3 Components Analysed

There are many html attributes that can be analysed and tested in relation to accessibility. For the purposes of this research, anchor, image, IFrame, heading, JavaScript and advanced rich internet application (aria) tags were collected. The individual page report shows the number of anchor, image and IFrame tags scanned and changed. If there was no heading level structure in the document, a single heading was inserted after the body. Warnings were logged for image tags with no height or width, and for aria and JavaScript documents.

4.2.4 Elements that were excluded

There was no way, to determine what the height or width of an image should be. JavaScript could not be evaluated correctly. As with any form of code, there is no standard way to name variables, or to know what they will do. As a result, it was not possible to determine whether the JavaScript will alter the document once it is activated. The same can be said for aria roles. There is no way to accurately predict what effect these will have on the document. Internet Explorer and Firefox handle them differently.

4.3 Conducted experiments

Three sets of data were analysed. Irish educational sites and a random set of 100 sites from Ireland and Spain were selected by a Google search. Spain is one of the few countries that did not have to migrate a lot of legacy systems to web environments. As a result, it was thought, that their levels of accessibility should have been better than most. In each of these data sets, only the home page of each site was evaluated. Although studies have shown that the home page is not always a good way to evaluate a web site, it was considered to be indicative of what the rest of the site would be like with regards to accessibility. As the method being used was quantitative, only one page per site was used. To analyse sites with large and small numbers of pages would give rise to figures that could not be matched, when extracting averages and such. There could also be instances, where sites have a single page modified and changed by JavaScript. If this was compared to a similar site with multiple pages, the figures would be totally

Groups	Count	Sum	Average	Variance
Irish Edu complete Sites	66	1199	18.1666	210.2333
Irish complete Sites	100	2000	20	192.4444
Spanish complete Sites	100	2139	21.39	188.0988

TABLE 4.1: Summary of Anova test on complete identified tags

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	413.8252	2	206.9126	1.0599	0.3479	3.0301
Within Groups	51338.9566	263	195.2951			
Total	51752.7819	265				

TABLE 4.2: Summary of Anova test on complete identified tags

different. Each set of data was run through the windows version of WAAM. For each page, a modified version of that page was created as well as a report showing what was modified and what could not be changed. The percentage of changes versus what was scanned was also recorded. An overall site report was produced, showing statistics for each page, and overall statistics for the test. This is similar to the page report, but includes the average, minimum and maximum number of changes versus items scanned. The three sets of data were compared. This showed the compliance rate of each selection, in relation to accessibility concerns for blind users.

The following subsections evaluate the sample data collected using an ANOVA approach. The Null Hypothesis associated with ANOVA calculation is that there does not exist a significant difference between the underlying populations from which the samples had been drawn from. The Alternative Hypothesis being that the sample data, in each case, seem to characterise different underlying populations.

4.3.1 Complete Data Set analysis

Table 4.1 below depicts each samples overall statistics. For example, row 1 of table 1 shows the descriptive results for the Irish Educational Site analysis. From this we can see that 66 sites were analysed with an average of 18% of image tags being scanned requiring change.

Table 4.2 depicts the results of the actual ANOVA F-test. From the application of the ANOVA procedure to the complete set across all three samples, the statistic confirms at a 5% significance level that there is insufficient evidence to believe that the samples had

Groups	Count	Sum	Average	Variance
Irish Edu Image Sites	66	2402	36.3939	1326.7039
Irish Image Sites	66	2284	34.6060	1054.8885
Spanish Image Sites	66	2495	37.8030	924.7759

TABLE 4.3: Summary of Anova test on Image tags

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	338.8585	2	169.4292	0.1537	0.8576	3.0422
Within Groups	214913.9545	195	1102.1228			
Total	215252.81313	197				

TABLE 4.4: Results of Anova test on Image tags

Groups	Count	Sum	Average	Variance
Irish Edu iFrame Sites	66	1200	18.1818	1510.4895
Irish iFrame Sites	66	2175	32.9545	2214.5979
Spanish iFrame Sites	66	3247	49.1969	2344.5606

TABLE 4.5: Summary of Anova test on iFrame Sites

been drawn from different populations. In particular, ($F = 1.05$, $p = 0.347$). Considering that there was no significant different between the different groups when we consider the complete tags, further analysis was required to identify the relationship between the individual tags.

4.3.1.1 Analysis of variance for the Image Tags

Table 4.3 above depicts each samples overall statistics. For example, row 1 of table 1 shows the descriptive results for the Irish Educational Image Site analysis. From this we can see that 66 sites were analysed with an average of 36% of image tags being scanned requiring change.

Table 4.4 depicts the results of the actual ANOVA F-test. From the application of the ANOVA procedure to image tags across all three samples, the statistic confirms at a 5% significance level that there is insufficient evidence to believe that the samples had been drawn from different populations. In particular, ($F = 0.15$, $p = 0.86$).

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	31767.7676	2	15883.8838	.8508	0.000525	3.0422
Within Groups	394527.1212	195	2023.2160			
Total	426294.8889	197				

TABLE 4.6: Results of Anova test on iFrame Sites

Groups	Count	Sum	Average	Variance
Irish Edu href tags	66	807	12.2272	129.5013
Irish href tags	66	1043	15.8030	199.9452
Spanish href tags	66	1003	15.1969	150.3759

TABLE 4.7: Summary of Anova test for href tags

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	483.3939	2	241.6969	1.5111	0.2232	3.0422
Within Groups	31188.4697	195	159.9408			
Total	31671.8636	197				

TABLE 4.8: Anove test on complete tags

4.3.1.2 Analysis of variance for the iFrame Tags

Table 4.6 depicts the results of the actual ANOVA F-test. From the application of the ANOVA procedure to iFrame tags across all three samples, the statistic confirms at a 5% significance level that there is sufficient evidence to believe that the samples had been drawn from different populations. In particular, ($F = 7.85$, $p = 0.0005$). Further analysis of the raw data indicates that Spanish sites have a much higher volume of utilisation but also have a high rate of inconsistency when considering accessibility for blind people.

4.3.1.3 Analysis of variance for the anchor Tags

Table 4.8 depicts the results of the actual ANOVA F-test. From the application of the ANOVA procedure to href tags across all three samples, the statistic confirms at a 5% significance level that there is insufficient evidence to believe that the samples had been drawn from different populations. In particular, ($F = 1.5$, $p = 0.22$).

4.4 Conclusion

In conclusion, the Null Hypothesis associated with ANOVA calculation is that a significant difference between the underlying populations from which the samples have been drawn does not exist. However, it was also found that within the population identified, Spanish sites utilised iFrames at a greater consistency level, and 49% of them required changes to their iFrame tags to enhance accessibility.

Chapter 5

Conclusions and Future Work

5.1 Introduction

Accessibility has been, and will continue to be an issue for web environments all over the world. This research focused particularly on the needs for the blind internet user. Three sets of data were analysed. Irish educational sites and 100 random sites from Ireland and Spain were selected by a Google search. As shown in the data analysis, there is still a lot to be addressed. Even though research suggests that Spain may be slightly ahead in this regard, statistical analysis of the data does not agree with this. The research has also shown, that full automation of web site accessibility is not possible. Regardless of the tool being used, there is still the need for human intervention. The Web Assessment Accessibility Model (WAAM), produced modified web pages for the sites scanned. However, it only replaced what it considered to be invalid text with messages that would need to be checked manually. It was impossible to evaluate things such as the height and width of images, and the probability that JavaScript and aria roles would make the page once more, inaccessible. There is a place for tools such as this, but they do need to be closely monitored. The single advantage that WAAM has, is that it produces modified web pages, showing web developers, that it is easier than they thought, to make accessible web sites for blind users. The experiments were conducted in a controlled environment. Microsoft's Windows 7 home edition, internet explorer version 9, Mozilla's Firefox version 21, and the jaws screen reader version 14 from Freedom Scientific were used. The research focused on blind users, as to try to make web sites generally accessible was considered to be too large a task given the research

time. The method used was quantitative, as a properly controlled environment, and access to a valid number of blind users was not possible. The WAAM software was developed using Microsoft's Visual Studio 2012, which at the time, was considered to be the most suitable development. All critical software updates were implemented when the experiments were carried out. No updates were implemented during the research. All home pages scanned were downloaded. Scanning live pages was considered to be unpredictable, as there could be changes between tests.

5.2 Further work

As shown, the various studies conducted from 1999 until 2013, have generally focused on accessibility for all, using the various guidelines and standards. There is a great need for further study into this. However, the studies should focus on particular disabilities and their needs. Although this would narrow the field, the recommendations would be easier to implement. This research focused on blind users. As a result, developers could use WAAM to produce modified pages from their own sites. If this was also done for other disabilities, the various studies could be combined. General research has its place, but it is just that, general. The current version of the software only scans for six items in a web page. These are the anchor, image, IFrame, heading, JavaScript and aria role tags. With further work, this tool could be much more comprehensive, and produce grater results. If the aria roles were properly analysed, the tool may be able to determine what is required, and insert them, as it does with the heading levels. Other elements such as tables, span and area tags could be analysed. It would be an advantage to conduct a qualitative study using WAAM, to see how accurate the newly modified web pages are for blind users. These could be modified further, and scanned again. Using both algorithmic and judgement methods, may yield a more conclusive set of results. The model could be redeveloped on a more stable platform. Although Visual Studio 2012 was used, parts of it were not accessible. It also produces a lot more by default than required for this software. As noted, it was the only platform available at the time, that was appropriate for such a development. Hand-held devices were also mentioned during this research. However, they were not evaluated, as there is no single device that has a reliable screen reading package. Perhaps one of the most useful versions is that provided by Apple on their iPhone. However, this proved to be

unusable when tried. Nokia also have screen reading technology for a narrow range of their phones. It too, does not allow access to web sites. Currently, the model evaluates anything that is within the "htm" file that is scanned. It is entirely possible, that this file could contain irrelevant information such as style sheet information, and pieces of JavaScript. The model could be further enhanced to recognise this, and eliminate it. Great care would be required when doing this, as JavaScript can contain references to anchor image and iframe tags within it. These would need to be evaluated to provide comprehensive results. It would be interesting, to take greater samples of data from the various populations, and statistically analyse them. Three sets of data were analysed, but the samples were relatively small. To analyse a larger sample of data from the same sources, would add weight to the statistical analysis produced for this research.

5.3 Conclusion

The internet is here to stay. It has now become part of daily life. People are using it, whether they realise it or not. Making a mobile phone call, requires some sort of access to the internet. Online business is also on the increase. With the advent of the cloud, it is easier than ever, to have information held and accessed at minimum cost. As a result, even small businesses are taking advantage of the internet. There are a lot of companies offering facilities for people to build their own web sites. These require absolutely no i.t. experience, as the user is prompted by the various tools when building their sites. As a result, accessibility particularly for blind users will continue to decline. A lot of the tools being used currently to develop web sites, do not enforce accessibility standards, and in some cases, they themselves are inaccessible. JQuery UI and Adobe's Flash are prime examples. Larger companies such as Microsoft and Google also do not take the problem seriously. As mentioned earlier, a query opened with Microsoft in January of 2013, had received absolutely no serious consideration by July of the same year. Google's chrome browser is completely inaccessible to the jaws screen reader. Over half of the papers used in this research had to be converted from Adobe's pdf into another format. Although Adobe claims their reader is "accessible", in reality, it has a long way to go. It is worth noting, that the papers produced as accessibility studies, were themselves, inaccessible without conversion. Cloud providers such as Heroku and EngineYard have enforced no controls over what is deployed on their platforms. In reality, disabled people accessing

the internet are still a minority. It is unlikely, that this will change. Web developers will always write their applications with the majority of the population in mind. They will also take the easiest approach to web design, in order to produce web sites efficiently and quickly. As a result, accessibility will continue to suffer. Unless there is a major change in the legislation governing the implementation of web sites, there is very little that can be done to address this problem. Only with large amounts of training and awareness, can this situation be changed. Perhaps, if the larger companies such as Microsoft and Google were to take this seriously, other businesses and organisations would follow. At the time of this research, no hand-held device or mobile "app" could be found that was accessible to the blind user.

Appendix A

Data Generated for Image Tags

The following table represents the data collected for the image tags.

Irish Edu Image Sites	Irish Image Sites	Spanish Image Sites
100	83	25
79	98	9
88	30	18
33	0	29
67	55	88
4	0	18
0	41	2
0	100	30
61	76	45
29	10	35
0	22	55
0	20	14
73	19	1
72	40	64
0	8	8
97	0	66
38	43	43
47	50	99
0	7	37
43	96	13
0	0	75
0	8	35
10	62	5
95	89	90
73	77	3
0	32	19
0	0	27
14	5	6

Irish Edu Image Sites	Irish Image Sites	Spanish Image Sites
25	53	50
11	29	70
14	20	21
83	0	89
91	8	53
0	100	76
87	3	87
70	9	0
0	24	91
0	0	46
9	0	48
0	37	18
0	46	25
15	67	0
0	25	0
50	0	26
13	62	47
50	0	14
13	12	44
0	77	83
50	71	90
11	13	23
0	52	0
0	21	12
100	0	95
13	43	64
75	82	50
32	70	37
15	20	0
0	89	8
13	16	18
100	86	56
50	6	8
100	19	85
100	53	0
61	0	46
42	0	56
86	0	0

Appendix B

Data Generated for HRef Tags

The following table represents the data collected for HRef tags.

Irish Edu iFrame Sites	Irish iFrame Sites	Spanish iFrame Sites
0	0	100
0	0	100
0	0	0
0	0	75
0	100	100
0	100	0
0	100	0
0	0	100
0	0	100
0	0	0
0	0	50
0	100	0
0	0	0
0	100	100
0	0	0
100	100	100
0	0	0
0	0	0
100	0	0
0	100	0
0	0	100
0	100	0
0	100	0
0	0	100
0	0	0
100	0	80
0	100	0
0	0	25

Irish Edu iFrame Sites	Irish iFrame Sites	Spanish iFrame Sites
100	0	0
100	0	100
0	0	0
100	0	100
100	0	100
0	0	100
100	0	100
100	0	0
0	0	0
0	0	100
100	100	100
0	100	67
0	100	100
0	75	100
0	0	100
100	0	0
0	100	0
0	100	100
0	100	100
0	100	100
0	0	100
0	0	0
0	100	0
0	100	100
0	0	100
0	0	100
0	0	0
0	0	0
0	0	0
0	0	100
0	0	100
0	0	50
100	100	100
0	0	0
0	100	0
0	0	0
0	0	0
0	0	0

Appendix C

Data Generated for the iFrame Tags

The following table represents the data collected for the iFrame tags.

Irish Edu Href Sites	Irish Href Sites	Spanish Href Sites
58	4	25
0	26	8
5	6	0
18	15	17
11	21	17
19	8	0
14	17	15
3	13	8
5	8	11
11	6	5
0	15	0
0	16	26
12	8	2
10	26	12
37	22	18
0	6	35
11	5	2
23	5	27
4	2	11
12	20	29
9	35	19
5	6	1
3	32	0
25	59	33
12	31	7
13	4	13
12	22	18
5	11	64

Irish Edu Href Sites	Irish Href Sites	Spanish Href Sites
4	15	24
46	21	25
9	1	26
17	16	2
17	17	20
6	0	12
14	2	23
29	20	0
0	33	3
12	3	10
25	1	23
3	24	24
0	29	15
6	0	10
14	14	31
20	0	14
17	13	19
1	29	30
0	9	16
8	37	11
11	4	14
15	21	11
1	0	0
7	21	17
9	0	9
7	26	23
13	14	3
31	18	1
10	13	0
5	11	50
5	1	19
35	7	15
18	1	23
23	33	26
23	22	11
2	50	6
0	0	14
7	68	0

Bibliography

[1] V.L. Hanson and J.T. Richards. Progress on website accessibility? *ACM Transactions on the Web (TWEB)*, 7:1 – 30, 2013.

[2] B. Parmanto and X. Zeng. Metric for web accessibility evaluation. *Journal of the American Society for Information Science & Technology*, 56:1394 – 1404, 2005.

[3] S. Hacket and B. Parmanto. Homepage not enough when evaluating web site accessibility. *Internet Research*, 19:78 – 87, 2009.

[4] J. De Andres, P. Lorca, and A. B. Martinez. Factors influencing web accessibility of big listed firms: an international study. *Online Information Review*, 34:75 – 97, 2010.

[5] S. Sandhya and K.A.S Devi. Accessibility evaluation of websites using screen reader. *7th International Conference on Next Generation Web Services Practices (NWeSP)*, pages 338–341, 2011.

[6] M. Eyadat and J. Lew. Web accessibility factor a key focus for serving students. *Review of Business Research*, 11:80–87, 2011.

[7] K. Johari and A. Kaur. Measuring web accessibility for persons with disabilities. *4th International Conference on Computational Intelligence and Communication Networks*, pages 963–967, 2012.

[8] R. Goncalves, J. Martins, J. Pereira, M.A. Oliveira, and J.J.P Ferreira. Enterprise web accessibility levels amongst the forbes 250: Where art thou o virtuous leader? *4th International Conference on Computational Intelligence and Communication Networks*, 113:363–375, 2013.

[9] J. Bailey and E. Burd. What is the current state of web accessibility? *Proceedings of the Eighth IEEE International Symposium on Web Site Evolution*, pages 69–74, 2006.

[10] S.M. Bruyere, W. Erickson, and S. VanLooy. Information technology (it) accessibility: implications for employment of people with disabilities. 27:397–405, 2006.

[11] J. Craven and A. Nietzio. A task-based approach to assessing the accessibility of web sites. *Performance Measurement and Metrics*, 8:98–109, 2007.

[12] J. Karreman, Van Der Geest, and E. Buursink. Accessible website content guidelines for users with intellectual disabilities. *Journal of Applied Research in Intellectual Disabilities*, 20:510–518, 2007.

[13] M. Debevc, P. Kosec, and A. Holzinger. Improving multimodal web accessibility for deaf people: sign language interpreter module. *Multimedia Tools and Applications*, 54:181, 2011.

[14] D. Sato, H. Takagi, M. Kobayashi, S. Kawanaka, and C. Asakawa. Exploratory analysis of collaborative web accessibility improvement. *ACM Transactions on Accessible Computing (TACCESS)*, 3:1–30, 2010.

[15] R. Kuber, W. Yu, and M.S O'Modhrain. Evaluation of haptic html mappings derived from a novel methodology. *ACM Transactions on Accessible Computing (TACCESS)*, 4:1–28, 2011.

[16] G. Brajnik, Y. Yesilada, and S. Harper. The expertise effect on web accessibility evaluation methods. *Human-Computer Interaction*, 26:246–283, 2011.

[17] G. Brajnik, Y. Yesilada, and S. Harper. Is accessibility conformance an elusive property? a study of validity and reliability of wcag 2.0. *ACM Transactions on Accessible Computing (TACCESS)*, 4:1–28, 2012.

[18] S. Marquez, F. Moreno, J. Coret, E. Jimenez, F. Alcantad, and I. Guarinos. Web accessibility for visually disabled. evaluation of the inclusite solution. *15th International Conference on Interactive Collaborative Learning (ICL)*, pages 1–5, 2012.