Investigating How HTML5 Improves Accessibility

FATEMEH MIRZAE D NAEEENI
Investigating How HTML5 Improves Accessibility

Fatemeh Mirzaee Naeeni

Department of Computer and Systems Sciences

October 2016

Supervisor:
Associate Professor Fredrik Kilander

This master's thesis corresponds to 30 credits.
Abstract

Nowadays, a lot of activities such as social communications, educational courses, banking processes, different types of entertainment, online registrations and many more are done through Internet websites. Millions of users use these websites to maintain their daily needs.

The challenge in front is that these websites are getting more and more complex, and the need to have well-designed websites becomes more important and at the same time more difficult. Knowledge of usability and accessibility according to the user needs can help designing more convenient while competent websites. It is essential to know that users with different levels of ability may visit a website. They should be able to easily interact with the website based on their abilities, likes and dislikes. Users usually leave the website early if they find it difficult to handle. They may look after easier and more practical solutions to do what they need to do. So it is somehow important to design websites by considering all levels of abilities and disabilities. Therefore, there are standards that must be followed by designers. Basically, usability experts define standards for making a website operational for a vast range of users by reducing the sources of complexity which potentially lead to a problem. Standards set by experts for accessibility aspects mainly focus on making a website more approachable to a bigger portion of users including users with disabilities.

This thesis investigates the role of HTML5 markup language in improving the accessibility of the websites. To achieve that, the main factors of accessibility in designing websites are considered and based on that the evaluation has been performed. To gain more applicable results, this thesis concentrates on screen readers as a kind of assistive technologies.

The results illustrate that HTML5 markup language can improve the accessibility of a website design. Currently, there are still some difficulties to overcome, for example, browsers lack support for HTML5 and the screen readers are not as effective as they are needed to be. This probably means that we still need more research to find optimal ways of constructing websites with a perfect accessibility and usability for screen readers.

Keywords: HTML5, accessibility, usability, disability, screen readers
Sammanfattning

Numera finns en hel del aktiviteter såsom social kommunikation, utbildningar, bankprocesser, olika typer av underhållning, online registreringar och många fler som sker genom webbplatser på Internet. Miljontals användare använder dessa webbplatser för sina dagliga behov.


Denna rapport undersöker betydelsen av HTML5 med avseende på att förbättra tillgängligheten av webbplatser. För att uppnå detta, har de viktigaste faktorerna för tillgänglighet i utformning av webbsidor beaktats och grundat på det som riskerar att komplexitet som potentiellt leder till problem. Normer som fastställts av experter på tillgänglighetsaspekter fokuserar huvudsakligen på att göra en webbplats mer tillgänglig för en större del av användarna, inklusive användare med funktionshinder.

Resultaten visar att HTML5 kan förbättra tillgängligheten hos en webbdesign. För närvarande finns det fortfarande vissa svårigheter att övervinna, till exempel att webbläsare saknar stöd för HTML5 och att skärmöverläsare är inte lika effektiva som de behöver vara. Detta innebär förmodligen att vi fortfarande behöver mer forskning för att hitta optimala sätt att bygga webbplatser med perfekt tillgänglighet och användbarhet för skärmöverläsare.
Acknowledgements

I pay my sincere gratitude to all the people who made this thesis possible. Special thanks go to the academic supervisor, Fredrik Kilander for his time, constructive feedbacks and supports to form this thesis a true academic one.

Finally, on a personal note, I am conveying my heartiest thanks to my family and my dear friends for their inspiration and motivation.
Contents

Abstract .............................................................................................................................................. 3
Sammanfattning .............................................................................................................................. 5
Acknowledgements .......................................................................................................................... 6
1. Introduction ..................................................................................................................................... 9
   1.1. Background ............................................................................................................................... 9
   1.2. Aim of Thesis ............................................................................................................................ 10
   1.3. Problem Description ............................................................................................................... 10
       1.3.1. Research Questions .......................................................................................................... 12
   1.4. Method ..................................................................................................................................... 12
   1.5 Risks, ethics and sustainable development ............................................................................... 13
   1.6 Report overview ...................................................................................................................... 13
2. Method .......................................................................................................................................... 15
   2.1 First Steps ................................................................................................................................. 15
   2.2 Study Materials for the investigation ....................................................................................... 16
   2.3 WCAG Principles and Main Research Question ....................................................................... 16
3. Usability and Accessibility ............................................................................................................ 19
   3.1. Usability ................................................................................................................................... 19
       3.1.1. Usability Concept ............................................................................................................... 19
       3.1.2. Usability Goals .................................................................................................................. 20
   3.2. Accessibility ............................................................................................................................. 21
       3.2.1. Accessibility Principles Based on WCAG 2.0 ................................................................. 22
       3.2.2 Accessibility Checkers ....................................................................................................... 23
4. Concepts and Definitions ............................................................................................................ 27
   4.1. HTML ....................................................................................................................................... 27
   4.2. HTML5 ..................................................................................................................................... 27
   4.3. ARIA ......................................................................................................................................... 28
5. Assistive Technologies .................................................................................................................. 31
   5.1. Assistive Technologies Types ................................................................................................. 31
5.2. Screen reader ........................................................................................................32
6. HTML5 and Accessibility ..........................................................................................33
   6.1. HTML5 Semantic Tags ..................................................................................33
   6.2. How to Increase Accessibility ........................................................................34
      6.2.1. Keyboard Support ................................................................................34
      6.2.2. Focus Management ...............................................................................35
      6.2.3. Labeling .................................................................................................35
      6.2.4. ARIA ......................................................................................................35
7. HTML5 Accessibility Solutions ...............................................................................39
   7.1. Screen Readers Accessibility Problems .........................................................39
      7.1.1. Skip Links and nav Element ...................................................................39
      7.1.2. Tables .....................................................................................................40
      7.1.3. Forms .....................................................................................................42
      7.1.4. ARIA Project .........................................................................................42
   7.2. How HTML5 Improves Accessibility .................................................................43
8. Conclusion and Future Work ..................................................................................47
   8.1. Conclusion .......................................................................................................47
   8.2. Future work .....................................................................................................48
1. Introduction

1.1. Background

Louise Veling in his paper, “Why is user experience design (UED) important for organizational success?” [1], declares how important it is to involve users in our business and how this involvement may increase the value of business. Based on Veling’s observations, ignoring the user role and forgetting to involve and motivate the user to share experience with the business developers, leads to a failure of the business because a business needs to know its customers and their requirements.

On the other hand it is almost impossible to ignore the role of the World Wide Web in our everyday lives. It is what we can see around us. All companies and business owners have found that to continue, they have to have a website that can be reached not only on computers but also on other devices like mobile phones and tablets. These days it is very easy to find a product or service website that contains information and different features to access that product. In fact it is a common and popular way for different businesses to use Internet and IT as an interface between themselves and their customers.

Jeff Sauro in his article, “A brief history of usability” [2], speaks about the history of usability and human efforts to make things as user-friendly as possible. According to Sauro, professional usability started in the 1980s. Today having a user-friendly design mindset is so essential for developers. In order to design a good website that can provide enough information and useful services for users, it is a good idea to involve them in design and tests. It means that developers, for example, need to watch user interaction with their website prototype to see what problems they face in performing a task. Based on what they find in these observations, they may change and improve their design to something more useful.

Jonathan Lazar in his book, “User-centered web development” [3], mentions that the main web development challenges are different devices, different display sizes and also different browsers. Lazar discusses that a very essential difference between using a software and using a website or a web based application is that when users use a software, they are ready to read documents or get trained but it is not possible to train users for all possible website designs and users will never spend that amount of time for each website. It is interesting to know that in developing a usable and accessible website many sciences such as computer science, psychology, sociology, marketing and usability engineering are involved.

When the number of website users increases, their levels of abilities also differ. Based on Zimmermann in his paper “Beyond Usability—Measuring Aspects of User Experience” [4], depending on its subject and popularity, a website may have users of different cultures, different levels of education, different ages and different levels of abilities; that means users with some disabilities may visit the websites too. A successful design needs to meet different user groups need.

When web developers design a system for people with some kind of disability it is essential to be error free. There are different types of assistive technologies that help people
with disabilities to interact with websites. This is discussed in a separate chapter. This thesis works on a special type of assistive technologies, screen readers, which is a helpful tool for people who are blind, visually impaired, illiterate, or learning disabled and the people who cannot or don’t want to read screens by themselves, often in combination with other AT such as screen magnifiers. These users can use screen readers to read the screen for them and lead them to do what they intend to. The reason to choose this technology is author's personal interest to know more about it and also, finding interesting sources and test tools has been a motivator.

1.2. Aim of Thesis

The main goal of this thesis is helping to gain some improvements in design of more accessible websites and web applications in order to serve all people with any range of abilities including people with different levels of disabilities.

To reach the goal, this thesis has investigated HTML5 facilities to improve accessibility in website designs.

In order to achieve more sufficient results, the work in this thesis is narrowed down to investigating how HTML5 can help designing more accessible websites for screen readers that are a kind of assistive technologies for people with sight disabilities.

Hopefully the result of this thesis can help web developers to choose better technologies to make their websites more accessible.

1.3. Problem Description

Each of us in our daily lives may work with several websites to perform different activities. Is it always easy to use all of these websites? It is a familiar experience to struggle with a website that one does not know how to start or finish a task. Or, for example, the user cannot find a specific feature. Of course, it is not a desired experience.

Now imagine the situation that the user has a kind of disability. In the scope of this thesis it can be a user who is blind. This user may use an assistive technology, for example, a screen reader to interact with the website. This user has no idea about the page appearance. It is not important for her how eye catching and colorful a website interface is. What this user needs is a well-formed page structure that the screen reader can understand and interpret for the user. Reasonable orders and connection between elements are essential; otherwise screen reader gets lost. So the website not only needs to be usable for users without disabilities, but also needs to be accessible for the screen reader. In fact in this case the screen reader is the user of the website and the website needs to be accessible for this user.

A good accessible website page needs to follow some essential rules and also needs to be designed with certain tools and contain certain elements. Hubert in January 2006 published his paper "Accessibility and usability guidelines for mobile devices in home health monitoring" [5] and discussed some of accessibility rules that are explained later in this paper.
This thesis has tried to investigate how the HTML5 markup language improves the accessibility of websites, based on accessibility rules especially for websites accessible for screen readers. In order to achieve this, the main accessibility factors and rules need to be known. Then it is important to study how HTML5 can fulfill any of the accessibility requirements and how HTML5 helps obey the accessibility rules.

According to WCAG official web page, [6], basic principles of accessibility are:

1. Perceivable
   It means that a website content including information and functional components must be clear and perceivable for users. For this principle there are four guidelines:
   
   1.1. Text alternative must be provided for non-text content with the same purpose.
   1.2. Alternatives for time-based media must be provided.
   1.3. It must be possible to present content in different ways while retaining same information and structure.
   1.4. Foreground and background must be separated so that the user can see and hear the content more easily.

2. Operable
   All user interface components goals must be achievable and navigation in the website must be clear and simple and meaningful. For this principle there are four guidelines:
   
   2.1. All functions must be made possible by keyboard.
   2.2. Users must have enough time to understand and work with content.
   2.3. There are some known causes for seizures. In designing a website those known reasons for seizures must be avoided.
   2.4. Navigation must be clear and user must be able to find what she needs.

3. Understandable
   Content and functionality of user interface must be understandable. For this principle there are three guidelines:
   
   3.1. Text must be readable and understandable.
   3.2. It must be possible for the user to predict appearance and operations.
   3.3. Users must be kept away from making mistakes and it must be possible to correct mistakes.

4. Robust
   Website must be designed and developed robustly because it needs to be interpreted reliably by user agents like assistive technologies not only for current ones but also for the future ones.

   There are several screen readers in the market and are also available for free. Based on Dearman and Troung in their paper “Leveraging Proprioception to Make Mobile Phones More Accessible to Users with Visual Impairments” [7], most of them, despite many good improvements in this technology, still are not very easy to use. On the other hand many websites are not accessible enough for these screen readers to be read and used efficiently.
1.3.1. Research Questions

HTML5 as a markup language provides some facilities for developers to develop websites and mobile applications. Are these applications and websites as accessible as they should be? Of course in this point the role of developers and the way they think about accessibility in their designs cannot be ignored but HTML5 as a tool can be investigated.

Main questions which this thesis has tried to answer are: Does HTML5 meet the requirements for the development of an accessible website? In order to answer this question it is necessary to know what the main requirements of an accessible website are.

Looking into the accessibility principles and guidelines that are listed in section 1.3, it is obvious that some of them are directly dependent on web designers and content managers’ skills. Some others can be reached by facilities provided by used markup language. In the scope of this paper finding enough reasons to fulfill any of these requirements leads to conclude that HTML5 has improved accessibility. So the main question here is that:

Which principles and guidelines of WCAG 2.0 are addressed by HTML5? How HTML5 has made it easier for developers to obey these principles and guidelines?

1.4. Method

This work is based on wide literature studies started by working on documents and resources generally about HTML5 and usability by the author. After selecting specific users group, users with disabilities, it continued with accessibility’s studies and main rules and factors in designing accessible websites. The author decided to limit the work on a group of users who are blind, visually impaired, illiterate, or learning disabled and the people who cannot or don’t want to read screens by themselves and also chose screen readers as the assistive technology to gain clearer results. Finding interesting materials about how HTML5 has improved accessibility of web pages for screen readers was the main reason to focus on this group of users, which need screen readers to interact with websites. Exploring in many different resources including books, articles, and weblogs and so on, led the author to conclude that the main fundamentals for accessibility concepts are defined and accepted reasonably by the people who work in different fields in this area.

Reading more papers and books and articles helped get a clearer picture of the work and also helped define better research questions. Based on almost all the articles read about screen readers, it is really important and essential to have a good and correct document object model (DOM) for each of the web pages in a website. Because the screen readers interpret a web page only by going through elements of the DOM, with a bad model, screen readers and their users get lost in the web page without being able to perform desired tasks.

Studies of HTML5 resources make the need of study about ARIA element as a part of HTML5 required. ARIA is an acronym for Accessible Rich Internet Applications. It used to be an addition to HTML before HTML5 but HTML5 has it as a part.

Resources have been found in the virtual and physical libraries and also everywhere on the internet. The main resources have been found in KTH university library in Kista and main
campus and also KTH library website. Also lots of keywords and phrases such as usability, accessibility, HTML5, HTML5 and accessibility guidelines, screen reader, HTML5 and screen readers, screen readers and accessibility, ARIA, ARIA in HTML5, ARIA and accessibility, usability and accessibility differences and so on have been used to search in Google for materials. Some resources for literature study regarding the topic of this thesis have been weblogs, posts or articles found on Internet just by searching for these concepts since the authors believe that individuals’ experiences in the web development field are very valuable since many of the findings in the usability and accessibility area are based on observations and interpretations of different developers and experts working with real users. Of course other sources have been known books and articles from web development people or organizations that know more usability, accessibility and web development. Finding about WCAG 2.0 accessibility principles, guidelines and studying more about HTML5 accessible solutions make it possible to answer the main research questions: Which principles and guidelines of WCAG 2.0 are addressed by HTML5? How has HTML5 made it easier for developers to obey these principles and guidelines? WCAG 2.0 declares four principles and several guidelines for accessibility and HTML5 has some improvements in structuring a web page. So the work is continued by evaluating HTML5 solutions based on WCAG 2.0 principles and guidelines.

1.5 Risks, ethics and sustainable development

This master’s thesis has concentrated mainly on investigating how HTML5 can improve accessibility in developing webpages. The main goal and ambition of performing this research has been helping to satisfy users specially those groups of users with different types of disabilities. Better accessibility in websites, softwares and mobile applications lead to better social interactions for these users. Otherwise it is reasonable if they feel disconnected from the society whose interactions are going to be more and more based on electronic devices and networks.

From an economical perspective websites, softwares and mobile applications as products can make more money for their owners if they are better designed and developed. With more satisfied users it is more possible to earn more money.

Also better accessibility in more websites can hopefully lead to reducing paper waste and save more trees.

1.6 Report overview

Chapter 1, Introduction, contains some background information and problem description. Also it contains research questions and a short summary of the method.

Chapter 2, Method, contains more details about how the investigation has been performed.

Chapter 3, Usability and Accessibility, contains concepts of usability and its goals also definition of and details about accessibility, accessibility principles and also introduction of some accessibility checking tools.

Chapter 4, Concepts and Definitions, contains HTML, HTML5 and ARIA general concepts.
Chapter 5, Assistive Technologies, speaks about different types of assistive technologies specially screen readers.

Chapter 6, HTML5 and Accessibility, introduces HTML5 semantic tags and explains recommended approaches to increase accessibility using HTML5.

Chapter 7, HTML5 Accessibility Solutions, discusses HTML5 solutions for accessibility. It mostly concentrates on new semantic tags that address some of the known screen readers’ problems reading web pages. Also it contains a conclusion based on mentioned screen readers’ problems and HTML5 solutions to them.

Chapter 8, Conclusion and Future Work, discusses results and also suggests some future works in this area.
2. Method

2.1 First Steps

This work is based on wide literature studies by the author. These studies are started by working on documents and resources generally about different versions of HTML especially HTML5 and also usability concepts. Then with concentration on a specific user group, users with disabilities, it continued with accessibility studies which are mostly about how to make a website or software more accessible for people with different kinds of disabilities or low level abilities to use normal tools. Working on materials about accessibility in web development led the research to focus on the main rules and factors in designing accessible websites. Accessibility resources have a great focus on people with disabilities that are the main group of users for this thesis. Several types of disabilities were found interesting at the beginning but the author decided to limit the work on a group of users who are blind, visually impaired, illiterate, or learning disabled and the people who cannot or don't want to read screens by themselves, to gain clearer results.

Exploring in many different resources including books, articles, and weblogs and so on, led the author to conclude that the main fundamentals for accessibility concepts are defined and accepted reasonably by the people who work in different fields in this area. Repetition of the same definitions and almost the same explanations for different concepts such as accessibility itself and reasons to have more accessible websites in order to serve people with different types of disabilities, show that how widely these concepts and requirements are accepted. Since all of these make a good reasonable sense, reading more and more materials about the mentioned concepts helped get a clearer picture of the work and also helped define better research questions.

By knowing what exactly accessibility means, why it is important and what the main rules to develop an accessible web page are, investigation on how HTML5 functions as a markup language can improve accessibility become much easier.

A wide study about assistive technologies by the author, led the research to screen readers that are a kind of assistive technologies used by people with sight disabilities. Finding interesting materials about how HTML5 has improved accessibility of web pages for screen readers was the main reason to focus on this group of users that needs screen readers to interact with websites.

Based on almost all the articles read about screen readers, it is really important and essential to have a good and correct document object model (DOM) for each of the web pages in a website. Because screen readers can interpret the web page only by going through elements of the DOM, with a bad model, they will get lost and as a result the user may get lost in the web page without being able to perform the desired tasks. So studying about document object model of a web page and also rules to develop a good model became essential in order to do a better investigation.
Studies on HTML5 resources led to studying about ARIA element as a part of HTML5. ARIA is an acronym for Accessible Rich Internet Applications. It used to be an addition to HTML before HTML5 but HTML5 has it as a part. Works continued by studying resources which are about how ARIA can help design more accessible websites.

2.2 Study Materials for the investigation

All the resources have been found in the virtual and physical libraries and also everywhere on the internet. The main resources have been found in KTH university library in Kista and the main campus and also KTH library website which makes it possible to explore the best databases for books, articles and magazines. Also lots of keywords and phrases such as usability, accessibility, HTML5, HTML5 and accessibility guidelines, screen reader, HTML5 and screen readers, screen readers and accessibility, ARIA, ARIA in HTML5, ARIA and accessibility, usability and accessibility differences and so on have been used to search in Google search engine to find materials. The author believes that concepts such as usability and accessibility are very high-level concepts that can be explained by individuals who are working or doing research in these fields from different perspectives. In other words, based on author’s few years of work experience, any person who works in this area may have a noticeable point of view about users and their interactions with a website or software. So some resources for literature study regarding the topic of this thesis have been weblogs, videos, posts or articles found on Internet. Of course other sources have been known books and articles from people or organizations that know more about usability, accessibility and web development. The interesting fact about these different resources is that they have been complementing one another and not conceptually contradicting one another.

The author has studied over one hundred resources but among those only less than seventy are referred to in the text while others have been improving the author knowledge about the related concepts and the problem.

2.3 WCAG Principles and Main Research Question

W3C, the main international standards organization for the World Wide Web, published a series of web accessibility guidelines through WIA, Web Accessibility Initiative. WCAG, Web Content Accessibility Guidelines, is a part of that series whose second version is released in December 2008. [21] These guidelines are developed and introduced by W3C in cooperation with people and organizations around the world in order to provide a shared standard for web accessibility. WCAG goal is to make web content more accessible for people with disabilities and it provides guidelines for natural information such as texts, images, sounds and also code that defines structure of a web page. The author decided to study and work based on WCAG 2.0 since it was developed and introduced by W3C and it has been popular and well accepted by accessibility experts and web developers.

According to WCAG official web page, [6], basic principles of accessibility are:
1. Perceivable
   It means that a website content including information and functional components must be clear and perceivable for users. For this principle there are four guidelines:
   
   1.1. Text alternatives must be provided for non-text contents with the same purpose.
   1.2. Alternatives for time-based media must be provided.
   1.3. It must be possible to present content in different ways while retaining the same information and structure.
   1.4. Foreground and background must be separated so that user can see and hear content easier.

2. Operable
   All user interface component goals must be achievable and navigation in website must be clear, simple and meaningful. For this principle there are four guidelines:
   
   2.1. All functions must be made possible by keyboard.
   2.2. Users must have enough time to understand and work with content.
   2.3. There are some known causes for seizures. In designing a website those known reasons for seizures must be avoided.
   2.4. Navigation must be clear and user must be able to find what she needs.

3. Understandable
   Content and functionality of user interface must be understandable. For this principle there are three guidelines:
   
   3.1. Text must be readable and understandable.
   3.2. It must be possible for user to predict appearance and operations.
   3.3. Users must be kept away from making mistakes and it must be possible to correct mistakes.

4. Robust
   Website must be designed and developed robustly because it needs to be interpreted reliably by user agents like assistive technologies not only for current ones but also in the future.

Note: These principles and guidelines list is provided several times in this paper to make it easier for the reader to reach them.

Finding about WCAG 2.0 accessibility principles and guidelines and studying more about HTML5 accessible solutions, make it possible to answer main research questions:

Which principles and guidelines of WCAG 2.0 are addressed by HTML5? How HTML5 has made it easier for developers to obey these principles and guidelines?

WCAG 2.0 declares four principles and several guidelines for accessibility and HTML5 has some improvements in structuring a web page. So the work is continued by evaluating HTML5 solutions based on WCAG 2.0 principles and guidelines. This thesis has tried to investigate how the HTML5 markup language improves the accessibility of websites, based on accessibility
rules especially for accessible websites for screen readers. In order to achieve this, main accessibility factors and rules need to be known. Then it is important to study how HTML5 can fulfill any of accessibility requirements and how HTML5 helps obey accessibility rules.

Based on WCAG 2.0 there are four accessibility principles and eleven accessibility guidelines. Going through the principles and guidelines, a website content including information and functional components must be clear and perceivable for users. The fact that information in a website needs to be clear and perceivable for the users isn’t something that HTML5 can take care of. In fact it depends how a developer decides about the website content. But to create a clear functionality a tool such as HTML5 can play a role. The possibility to add text alternatives or the possibility to present content in different ways can be investigated. Also it is possible to investigate how HTML5 supports using keyboard to navigate or to function. For a blind user mouse is not an interaction option.

The amount of time that user may need to understand content or do what he intends to do is something that the developer needs take care of. Also avoiding from known causes for seizures is out of HTML5 investigation scope. But the document object model that can be provided by HTML5 can be investigated and as a result the navigation improvement can be investigated. Readable and understandable text, possibility to predict appearance and operations and preventing user from making mistakes or providing him some approaches to correct his mistakes and also having a robust system are all responsibilities of the developer.

After analyzing the principles and the guidelines, the author decided to concentrate on those ones that can be investigated in terms of HTML5 accessibility solutions. The author has tried to investigate if HTML5 has improved document object model of a website and also if HTML5 has a good keyboard support and if the new elements of HTML5 has helped the readability of a webpage or not.

As a result, which is illustrated clearly in the chapter 7 of this paper, HTML5 has improved accessibility in web development based on some of the guidelines of WCAG 2.0.
3. Usability and Accessibility

In this chapter the most important concepts on which this thesis is based, are discussed. The author has summarized the most important concepts regarding usability and accessibility. Knowledge on usability and accessibility is essential to be able to understand this paper. Also the main goal of the thesis and the main questions on which the thesis is based are declared.

The main basic concept of this thesis is user centered web design with focus on users with sight disability. In other words, the question is how to design a website that meets these users’ requirements [3]. This thesis has tried to answer this question: Has HTML5 as a markup language helped developers design and develop more accessible web pages? And to prove the answer it has focused on screen readers and web pages DOM structure interactions.

The author believes that it is essential to start with understanding usability and accessibility concepts and later concentrating on HTML5 as the tool, which is selected to be analyzed according to the accessibility improvements that it can provide for web developers. Of course each of these can be connected to other concepts in different fields such as human computer interaction, psychology, some field of art and design and some more but discussing more details on these concepts is out of the scope of this thesis.

3.1. Usability

Goldberg and his co-workers in their paper “Usability and Accessibility in Consumer Health Informatics, Current Trends and Future Challenges” [8], state that the old computing was about what computers could do. The new computing is about what users can do. It illustrates how important the role of users and users satisfaction in the new computing is. May be because some of these technologies are advanced enough to meet their basic or even complicated goals, now it is essential to make them as rich as possible for the users to choose them among all others.

Bevan and his colleagues in their paper “What is Usability?” [9], define usability as “the ease of use and acceptability of a product for a particular class of users carrying out specific tasks in a specific environment”. Regarding web development, usability can be ease of performing desired tasks in a website. Finding any required information in the website scope, being able to use services which are provided in the website, meeting goals of it and having a good navigation experience by users and not getting lost or exhausted because of complication of workflow are some of what usability experts care about in evaluating a website. To keep focus on HTML5 and accessibility, this thesis has concentrated on main concepts of usability and especially accessibility and some of the main rules to have an accessible website.

3.1.1. Usability Concept

Based on Matt Jones and Gary Marsden in their book “Mobile Interaction Design” [10], people or in other words, users, are busy in performing many different activities in their daily lives. They need to work with different concepts and tools everyday not only in their private
lives but also at their jobs. It needs lots of energy and time to learn and use all of these; so when it is a technological term they use only those they really need to make their jobs easier or have something helpful for them. In fact a technology should be presented to them in such a way that using it makes them happy and satisfied and they should like spending time in learning and using that technology. It is important that a technology be in harmony with other resources that users work with. For example when users visit a website to register for a course, they don’t like to focus on anything but the main purpose of their visit otherwise they may prefer to go for all old paper works which they know. As Jones and Marsden explain it, interaction design is like creating a building. To have a good designed building it needs to be made ready for a comfortable and secure life.

Gutierrez and Barchino in “Usability Issues Confronting Mobile Devices as Internet Interfaces for General Purpose Navigation” [11], Bevan and Kirakowsky and their colleagues in “What is usability?” [12] And Torrente and Prieto in “Sirius: A heuristic-based framework for measuring web usability adapted to the type of website” [13], explain what user finds in the browser needs to be understandable, learnable, helpful to achieve goals and simple and even interesting enough to make one ambitious to come back again. It is not true only for websites that introduce a business but it is essential for those kinds of products, which are web based so that users access a web browser to use those products.

Based on Bevan in the published paper from his workshop lecture, ”Valid Useful User Experience Measurement” [14], usability can be defined as user satisfaction. It means that it is important to create a system that users can like it. It happens when user is happy with using the system to achieve different goals and is satisfied with the results. Also it is important to create a system that users enjoy to use. They accept the system as an intelligent entity that can memorize some information and can suggest some approaches and this interaction should give the user good feelings. Also users need to feel comfortable while using the system. They shouldn’t feel confused or annoyed and they need to trust the system and it only happens when a system works as it supposed to.

3.1.2. Usability Goals

There are some factors to determine quality of experiences on a website. Jadav in his book, “Designing usable Web interfaces” [15], among those factors, mentions some like: navigation should be easy, texts should be readable, content should be organized and the media and communication should be integrated in an efficient way.

Helen Sharp, Yvonne Rogers and Jenny Preece, in the second edition of their interesting book, “Interaction Design beyond human-computer interaction” [16], explain some of the main goals for usability:

- **Effectiveness**
  This factor refers to how much a product, in this case a website, is useful in the field that it is developed for and how much it can fulfil user needs.

- **Efficiency**
This factor refers to how much a website design helps users to perform their tasks. It is about the quality of a design and its capabilities to increase performance.

- **Safety**
  This factor refers to how a website protects its users in unwanted and dangerous situations. A good design needs to provide possibilities like restoring, recovery and undo.

- **Learnability**
  This factor refers to how much it is easy to learn and use a website.

- **Utility**
  This factor refers to what suitable and useful activities are possible in a website that helps users to perform their required tasks and achieve what they need. A good design needs to have different possibilities to simplify and speed up user’s desired tasks in a way that the user likes it. For example a design can contain shortcuts that make a long and complicated task shorter and easier.

- **Memorability**
  This factor refers to how easy it is to remember the way in which the user can perform a task in a website.

Jonathan Lazar in his book "User-centered Web Development" [3], explains that usability testing is not functionality testing and it is not about code. It is the kind of test that tries to discover if a website is easy for users or not. In fact users are the leaders here. System developers cannot predict user interactions but they have to use their users to do a real test on their design. Many different situations need to be considered to develop a website or any web application. These tests need to be performed to ensure that there are no serious usability problems that will affect the user interaction.

### 3.2. Accessibility

Connor in his book “Pro HTML5 Accessibility” [17] explains that "Web accessibility means that people with disabilities can use the Web. More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web. Web accessibility also benefits others, including older people with changing abilities due to aging". Based on Connor, in order to have an accessible website web developers need to consider the different levels of people’s abilities. Some people may have sight disabilities, so they may not see what is going on a webpage or for some others sounds can be not usable or some users may have motion problems or maybe they cannot use a mouse or a keyboard to interact with a website.

Sulaiman och Masitah Ghazali2 in “A Systematic Literature Review of Accessibility and Usability Framework for Disabled Users” [18], explain that in order to use information technology, both usability and accessibility are very important qualities to help users. Concepts such as time of execution, performance, user satisfaction, security, efficiency, effectiveness and learnability are referred by "usability", and accessibility that often focus on people with disabilities can be defined as "the ability to access" the system and making it possible for the
user group to access a system, by using their assistive technology. Usability and accessibility are about users and interactions with designed systems.

Based on International Organization for Standardization (ISO) standard for usability [8], all users of a system must have an equivalent experience and it should not matter what assistive technology they use on a computer or mobile device. In designing a system it is essential to think about accessibility and usability. It means that web developers need to remember their user groups all the time and change the system as much as needed for the user groups to make it possible for them to use the system without any difficulties. This is what is called "user centered design".

The Authors of "Evaluation of Web Browsing Experience by People with Cognitive Disability" paper [19], state that Usability and Accessibility both have the same goal to improve effectiveness, efficiency and user satisfaction. Improving usability focuses on people whom a system is being built for and improving accessibility increases the number of people who can access it successfully. It makes it much easier to understand the relationship between usability and accessibility. Based on Krug in his book, "Don’t make me think: a common sense approach to web usability" [20], if a website is not accessible it is not usable at all.

3.2.1. Accessibility Principles Based on WCAG 2.0

W3C, the main international standards organization for the World Wide Web, published a series of web accessibility guidelines by WIA, Web Accessibility Initiative. WCAG, Web Content Accessibility Guidelines, is a part of that series whose second version is released in December 2008. [21]

You have seen these principles and guidelines previously in this paper but since those are very essential for this work the author decided to repeat them. According to WCAG official web page [6] the basic principles of accessibility are:

1 Perceivable

It means that a website content including information and functional components must be clear and perceivable for users. For this principle there are four guidelines:

1.1 Text alternative must be provided for non-text content with the same purpose.
1.2 Alternatives for time-based media must be provided.
1.3 It must be possible to present content in different ways while having same information and structure.
1.4 Foreground and background must be separated so user can see and hear content easier.

2 Operable

All user interface components goals must be achievable and navigation in website must be clear, simple and meaningful. For this principle there are four guidelines:

2.1 All functions must be made possible by keyboard.
2.2 Users must have enough time to understand and work with content.
2.3 There are some known causes for seizures. In designing website those known reasons for seizures must be avoided.
2.4 Navigation must be clear and the user must be able to find what she needs.

3 Understandable
   Content and functionality of user interface must be understandable. For this principle there are three guidelines:
   3.1 Text must be readable and understandable.
   3.2 It must be possible for user to predict appearance and operations.
   3.3 Users must be kept away from making mistakes and it must be possible to correct mistakes.

4 Robust
   Website must be designed and developed robustly because it needs to be interpreted reliably by user agents like assistive technologies not only for current ones but also for the future ones.

   Increasing web accessibility has some benefits. We can categorize these benefits into social, technical, financial and design factors. By improving accessibility we can provide possibility of communication and interaction in a better way for people with disabilities. Also we can improve development and the developer’s knowledge about user groups as well as quality of our systems. Better accessibility means less maintenance time and less server load.

   From the financial perspective, less maintenance time and less server load and improved search engine optimization are benefits of better accessibility.

   Connor, the author of “Pro HTML5 Accessibility” [17], also believes that better accessibility means better design. It means thinking about a large user group and thinking out of the box. It means that if developers think of users with different types of disabilities they probably can design and develop more accessible products.

3.2.2 Accessibility Checkers

   W3C organization website [22], introduces some helpful and interesting tools to check and evaluate accessibility of websites. At the moment of writing this paper which is October 2016, there are thirty-three tools presented in the W3C website. It is simple to try any URL in each of these tools and see the result.

   One of the most popular websites with a huge number of users is Google search engine. In the following figures the Google home page is checked by some of these tools. These figures are inserted in the paper to show the tools main interfaces so there is no need to read the text shown in them.

   WAVE is a set of tools, which can be added to Firefox and Google Chrome browsers as extensions. It has a good graphical user interface which makes it very easy for users to evaluate their pages.
In figure 1, the left side menu shows errors and warnings and other important information about the evaluated page. By hovering on each of these items, the related element, if it is visible, gets highlighted on the right side area. Also the user can see the source code and by double clicking on each item of the left side menu, related code lines get highlighted. It can be very handy and useful to determine mistakes and potential problems of a web page design regarding accessibility.

![WAVE accessibility check tool](image)

Figure 1 - WAVE accessibility check tool

AChecker is another web accessibility checker from W3C. AChecker has a good configuration panel in which the user can select guidelines to check the HTML code against. Another facility of AChecker is that the user can import a HTML file or even a piece of markup code. Provided result of AChecker contains details based on different rules of accessibility with code getting checked against. These details are categorized into known problems, likely problems, potential problems, HTML and CSS validations. Figure 2 shows AChecker configuration panel. It accepts a website address or an HTML file or even a piece of HTML code. User can determine validators and also guidelines version and levels to validate the web page against.

---

2. [http://achecker.ca/checker/index.php#output_div](http://achecker.ca/checker/index.php#output_div)
Figure 2 - AChecker configuration panel
Figure 3 shows the result of validation based on what the user has selected. The shown result in the picture is for Google home page validation based on WCAG 2.0 guidelines. Based on WCAG 2.0 guidelines, all used images need to have alternative texts. It is what that is missing in the Google home page as detected by the AChecker. AChecker also detected two more problems both of which are related to font colors. Google home page has a very simple design. As it is clear here, even this simple page is not as simple as it should be for screen readers to be read. The author of this paper believes that it shows how essential accessibility studies are.
4. Concepts and Definitions

4.1. HTML

Based on W3Schools definition [23], HTML, Hyper Text Markup Language, is a markup language made of tags, which can be used to construct a web page structure.

Mozelius in his book “Interactive web applications” [24], explains that web development is over twenty years old now. During all these years HTML has been the main tool to develop web pages. HTML is a simple markup language that even one doesn’t have to be a professional web developer or programmer to be able to create a web page using it. HTML is very easy and it has been a good reason to make web this much popular.

4.2. HTML5

Based on MacDonald in his book “HTML5: The Missing Manual” [25], HTML5 is the latest version of HTML. HTML as a markup language has been the main web development tool during the past 20 years.

Musti in his paper “HTML5 - An opportunity for Innovation” [26], explains that one of the most important characteristics of HTML5 is that it can give a very rich user interaction experience. This factor makes it very easy for developers to move faster in the market by a write it once and use it everywhere approach.

Online resource "HTML5 in MobiLe Development" [27], declares that introducing HTML5 has led to some important improvements in Javascript, CSS and also browsers. It means that now much more powerful browsers are needed to support HTML5. These browsers need to support playing video/audio directly without installing any plugin. Also they need to be able to store data locally. Visual effects and 3D graphic are richer in HTML5. Of course some features of HTML5 still don’t work on any browsers and some only work on some of the browsers and some Javascript libraries are needed to make some browsers to work with HTML5.

MacDonald in first edition of his book “HTML5: The Missing Manual” [25], also declares HTML5 has tried to be loyal to some older principles. Although it has led to great changes in browsers, Javascript and CSS technologies, however as a standard, it shouldn’t change used rules of web development. It means that it shouldn’t make developers to redevelop their pages. Old techniques have to remain because many average developers still use them and they will not migrate easily. Any change in the old rules has to have a good reason.

Curran and Bond and Fisher in their article “HTML5 and Mobile Web” [28] and Adams and his colleagues in their paper "HTML5 Case Studies : Case studies illustrating development approaches to use of HTML5 and related Open Web Platform standards in the UK Higher Education sector." [29], explain that by adding new features such as <video>, <audio>, <header> and <canvas> elements and also integration of SVG content which replaces the use of generic <object> tags, HTML5 has improved usability of web pages which are developed using it compared to HTML4. In HTML5 there is no need to install and use plugins and APIs to
include multimedia and graphical content on websites or web applications as it used be for HTML4. Also the semantic content of documents are richer by new elements such as `<section>`, `<article>`, `<header>` and `<nav>`. Semantic content of a web page is very important especially when user has to use an assistive tool like screen readers. Because as it is mentioned before, screen readers follow the document object model of a web page which in fact is a semantic model of the page.

Steve Faulkner in his article, "html5-accessibility-challenges" [30], describes how HTML5 by introducing new form controls and new browsers is leading web development to more standard approaches than before. One of the most important results of this is better accessibility in websites and mobile and web-based applications developed by HTML5 compared to those developed by the older versions of HTML.

### 4.3. ARIA

W3 organization [31], defines ARIA as: Accessible Rich Internet Applications (ARIA) is a web development technology that helps to have more accessible web pages. Web developers can add a set of accessibility attributes of ARIA to their HTML code. Based on W3 [32], HTML5 has `role` attribute that can define general type of an object. It can be something like `article`, `slider` or `alert`. Roles can be used to identify types of widgets and page structure. State of interface elements and dynamic regions can be identified using ARIA. Also ARIA can add descriptions for forms or for a component like a progress bar and it can provide keyboard navigation in different regions.

Featherstone in his website [33] and also Mozilla website [34], explains how ARIA helps web designers to add meaningful structure to their web content that makes it possible for screen readers to have a better understanding of the interface to help users with disabilities. As we discussed previously, screen readers discover page structure by interpreting document object model of the page. HTML5 by allowing a meaningful structure of the page makes it much easier for screen readers to interpret the page.

Based on W3C, World Wide Web Consortium, [32] [35], since advanced interfaces can contain different numbers and kinds of controls, good interaction with these controls is essential for assistive technologies. At the time of writing this report, many of designed websites are really poor in providing interaction possibilities for assistive technologies. Many of the functionalities are only available by using mouse and not keyboard (e.g., drag and drop or navigation). One important concern is that people with sight disabilities have problem to be informed when content gets updated by different web development technologies. ARIA in HTML has come to stage to help assistive technologies to gain more information from page content by providing better structure in DOM model of web pages. ARIA provides a structure for web interface. With ARIA attributes all features on the interface can be identified and can have relative situations to one another and also their current state can be determined. By using ARIA, interface developers can identify different regions on a web page such as menus, primary content, secondary content, banners etc. They also can make these different regions accessible by keyboard for people who cannot use mouse.
Petrie and Kheir in their paper "The relationship between accessibility and usability of websites" [36], state that ARIA intention is being used by screen readers to be able to interpret website structures and web pages for their users and therefore ARIA can improve accessibility for users of screen readers.
5. Assistive Technologies

Based on Wyke in his paper "User Centered Design for persons with disabilities How persons with cerebral palsy can be included in the design process" [37], what many people may think and believe is that input tools such as keyboard and mouse provide the fastest and the best way to use computers. The fact is that these tools as much as are easy for many people are not suitable for many others with different kinds of disabilities. Some kinds of disabilities make it very hard or even impossible to use these devices.

Based on Mauri and his colleagues in their article "On the Assessment of the Interaction Quality of Users with Cerebral Palsy" [38], the following factors are essential to consider and evaluate when choosing the exact mode of interaction:

- Mental capability: Some mental illnesses lead to cognitive attention, memory or other similar problems.
- Sensing and moving capability: Some limitations in sensing and moving reduce the number of devices that can be used.
- Previous knowledge: There are different tools to help people with some kinds of disabilities. For example some users with cerebral palsy know how to control a wheelchair by a joystick, and this knowledge can be a disadvantage when we want them to use this tool to control a computer.
- Knowledge improvement: When users interact with a system for a while usually there is a progress in their knowledge. In fact they learn how to use the system gradually. So evaluations must be able to accurately assess their progress.

Sometimes a mouse or a keyboard not only do not help a user to interact with a web page but also is totally useless.

5.1. Assistive Technologies Types

Assistive technology focused on providing products that are designed in order to serve people with different kinds of disabilities with more accessibility.

At the time of writing this paper, different kinds of assistive technologies are available. Based on Steve Faulkner in his blog "Assistive Technology Products" [39], some of them can be listed as below:

- Alternative input devices (Alternative keyboards, Electronic pointing devices, Sip-and-puff systems, Wands and sticks, Joysticks, Trackballs, Touch screens)
- Braille embossers
- Keyboard filters
- Light signaler alerts
- On-screen keyboards
- Reading tools and learning disabilities programs
- Refreshable Braille displays
- Screen enlargers, or screen magnifiers
- Screen readers
• Speech recognition or voice recognition programs
• Text-to-Speech (TTS) or speech
• Talking and large-print word processors

In this thesis we have focused on screen readers that are mostly useful for people with sight disabilities.

5.2. Screen reader

Interesting post with title "How well do screen readers support Web accessibility guidelines?" [40], explains that screen reader is a software that interprets texts and graphics and converts that to speech. A screen reader verbalizes everything on the screen. In the ideal situation it can tell the user about the page structure and all components, control buttons, menus and other elements of the page. Also it can tell the user what is possible to do by a component that is focused.

Screen reader is a technology of reading screen for a person who cannot or doesn't want to read it by themselves, for example, because of blindness or learning disability or other problems.

In fact a screen reader is a kind of software that can convert a graphical and text-based user interface to speech. It first interprets the page and creates a general structure of what can be read to user to make it possible to perform required functions, and then read it for the user.

Borodin and his colleagues in their article "More than Meets the Eye: A Survey of Screen-Reader Browsing Strategies" [41], explain that as long as web developers focus on developing their sites based on sighted people, web browsing using screen readers is hard and frustrating. Web designers may use many visual elements in their designs that are not accessible for people with sight problems. Usability and accessibility problems in a website can make web browsing with screen readers really hard especially for users who still don’t have any browsing strategy with screen readers that they use. When developers add an image in their designed page and forget to add an alternative text, in fact they make that image totally inaccessible for screen readers. When they add a widget to their website that only accepts mouse keys as input device, they make it inaccessible again. The other problem is more related to usability. Sometimes it is possible for screen reader to read content of a page but the page contains for example a long list of links or too many elements that even a sighted person may find it difficult to follow. Screen reader adds some more details about the elements while reading the page. So it will be a long text that makes it useless and confusing.

Based on Petrie and Kheir "The relationship between accessibility and usability of websites" [36], there are many advanced screen readers in the market but still the speed of web development technologies is much more than the speed of improvement in screen readers. Although today screen readers can update their off-screen buffers and cover updated contents affected by, for example Javascript, they still need much more improvement to understand semantics that ARIA of HTML5 has added to dynamic contents. There is a big gap between what web developers can provide for sighted users and what screen readers can present for users with sight disabilities.
6. HTML5 and Accessibility

This thesis main focus is on HTML5 and screen readers. Based on Faulkner in his blog post "HTML5 Accessibility Chops: section elements" [42], screen readers face some accessibility problems in web pages that make interpretation of the page complicated. In fact for a user with sight disability it doesn’t matter how colorful or beautiful a web page design is. The user cannot see it. This user can only interact with the page using an assistive technology such as screen reader. What is important is the page structure in the document. It is all about which tags are used and how they are connected to one another. If developers don’t take it seriously and concentrate only on page view, they will make screen readers face huge problems and make their pages inaccessible for users with sight disabilities using screen readers.

Based on Lazar and his colleagues in their article "What Frustrates Screen Reader Users on the Web:A Study of 100 Blind Users" [43] and Verma and his colleagues in their paper "A Framework for the Next Generation Screen Readers for Visually Impaired" [44], some of the older screen readers used to buffer web pages and then start to read the buffered pages for their users. But what about dynamic updates on the page? There are web pages with some sections with dynamic contents. Screen readers need to be informed of these dynamic updates to read it for their users. At the same time screen readers need to know if there is a dynamic update, which is not important. It is usually not easy to interact with a web page that user cannot see it. So it is important to omit irrelevant information. It is required to find a way to distinguish between important and unimportant contents too.

Also error messages, which appear in dialog boxes or as a text in another color, for example form validation errors are important to be announced to users.

In this chapter, HTML5 and ARIA features that have improved accessibility are discussed.

6.1. HTML5 Semantic Tags

Based on W3C [45], semantic elements are those tags that have a meaning. For example <div> tag is not a semantic tag because it doesn't give any information about its content. Some other tags such as <table> declare what their content is and how it may look like.

HTML5 introduces new semantic tags. Using these elements, the developer can create a semantic structure in her web page in which the different areas are well defined. In fact using HTML5 semantic tags, the developer can have element groups in different regions of the page. Labels and elements are connected better and applying styles is much easier than older versions of HTML.

HTML5 semantic tags as are defined in W3C, [45], as:

- <article>
- <aside>
- <details>
- <figcaption>
6.2. How to Increase Accessibility

Some of the most important factors in accessibility are keyboard support, focus management, labeling and using ARIA that is discussed briefly in this chapter.

6.2.1. Keyboard Support

As it is explained in the previous chapter, an accessible web design needs to consider a bigger user group that contains users of different ages, sexes and different levels of abilities.

Designing a website which works only or mostly with mouse and without keyboard support is far from accessibility.

This thesis main focus is on people with sight disabilities. These users may be completely blind or may have low visions. They need to use screen readers to interact with a website. A user with sight disability who uses a screen reader cannot use mouse. This user needs to use a keyboard to interact with web pages. It is enough to close the eyes and try to perform some tasks or even navigate in a website to test it.

Based on Peterson in his website post "Accessibility in HTML5" [46], considering screen readers as used assistive technology, keyboard support is essential to have an accessible website or web application. The main reason for it is that screen readers work only with DOM, document object model, of web pages.

Based on W3C [47], "The Document Object Model is a platform and language-neutral interface that will allow programs and scripts to dynamically access and update the content, structure and style of documents." To see DOM of a web page one can open Google Chrome browser developer tool add on. It shows hierarchy of tags. Also in Firefox and Internet Explorer browsers there are tools to show the DOM of a web page.

In fact DOM is what a screen reader works with. It is not important for the screen reader how a page looks like because it is not what its user needs to know. What screen reader needs is DOM. Then it tries to interpret the DOM for its users.

Based on Shearer in her youtube video, "Making Accessible Web Apps Using HTML5 and ChromeVox" [48], in a good design related elements are grouped. So screen reader doesn’t get lost in DOM. Using DIV and SPAN in designing a web page instead of using native HTML tags is a bad approach for screen readers because these elements don’t interact with keyboard events and are not focusable. These two factors are very important for screen readers. A control that
cannot respond to a keyboard event is not accessible. A button on a web page, which is clickable by mouse must respond to keyboard, for example, *Enter* key press too.

Peterson also explains how keyboard support is essential in order to have an accessible web page. Keyboard shortcuts are also very helpful because they are easy and fast to use and can make life much easier for users. Good documentation in this case is essential.

### 6.2.2. Focus Management

Shearer, [48], explains that focus management is an important factor to have more accessible web pages. Screen reader users who use keyboard to interact with web pages, use keyboard keys to move between page elements. So if an element is not focusable, for example a div or a span which is used to present a button or a link instead of using native HTML tags, the screen reader skips that element or gives its user incorrect and incomplete information based on other elements.

### 6.2.3. Labeling

Another important approach is labeling. All form elements and images in a website must have labels. It is because a user with sight disability doesn’t care about images on web page but she needs to hear a screen reader interpretation of those images and the screen reader looks into DOM to describe the page for its user.

Based on Shearer, a screen reader needs to find a description in *alt* attribute of an image. If it cannot find that attribute it will read the image file name that can be a meaningless text. So it is essential to add a description in *alt* attribute and also leave this attribute blank for unimportant images.

It is important to have a meaningful description in image *alt* attribute. HTML5 has improved accessibility for images compared to HTML4 by adding `<figure>` and `<figcaption>` tags using which the developer can add a caption for an image. The good thing here is that a screen reader can know what the exact caption of an image is. [46]

### 6.2.4. ARIA

HTML native tags have good support for the keyboard events and are focusable. So a good practice to design a web page is to use these tags as much as possible.

As it is mentioned before, using HTML native tags helps accessibility a lot but sometimes web designer needs more complex control that cannot be found in HTML native tags. In this situation the developer has to develop custom controls by using native controls. This can make a web page design complicated and reduce or ruin accessibility if the developer doesn’t care enough.

In this situation developer needs to make a custom control as much accessible as possible. The developer has to add important characteristics of native tags that make them accessible to the custom control. Same semantic is needed to be added. This new control needs
to be focusable and it must be accessible by keyboard. To add these two possibilities to a custom control, HTML5 suggests using ARIA. [48]

Based on W3C [49] definition, WAI-ARIA is “a technical specification that provides a framework to improve the accessibility and interoperability of web content and applications.”

Another definition for WAI-ARIA based on W3 [32] is “the Accessible Rich Internet Applications Suite defines a way to make Web content and Web applications more accessible to people with disabilities. It especially helps with dynamic content and advanced user interface controls developed with Ajax, HTML, JavaScript, and related technologies.”

The good news is that all modern browsers support HTML5 and for older browsers there are some solutions to teach them HTML5 support. Peterson in his website, [46], explains that HTML5 has promised to improve accessibility by its semantic elements and other stuff but the problem is that browsers technology is always some steps behind technologies such as HTML improvements. Of course W3C tries to keep this distance as low as possible but still there is a gap. It is true in HTML5 case too. Browsers have to update their infrastructure technologies according to the latest web development standards but it needs more time for supporting HTML5. The problem is that even if the latest versions of a browser support HTML5, some people use older versions, which don’t support it. So there must be another way to keep a web page that is developed by HTML5 accessible. It is where ARIA starts to play a great role.

ARIA is integrated into last version of HTML, HTML5. ARIA is a set of attributes that helps assistive technologies like screen readers to understand the document model of a web page.

As a matter of course W3C documents for ARIA and HTML5 suggest obeying some rules to use ARIA in web design. For example it mentions that the base practice is to use native HTML elements and attributes which are available instead of constructing new controls and adding ARIA roles to it. If the semantic and behavior that the web designer needs is not implemented or doesn’t support accessibility or if the developer needs a feature that is not available in HTML or if the developer cannot apply a desired style to a native element it is reasonable to develop a custom control.

ARIA has been around longer than HTML5; so browsers’ support for it is much better and also based on Shearer [48] some of the current assistive technologies support it better than HTML5 too. A web designer can develop a web page semantic code using HTML5 and ARIA as an add-on to make the page more accessible for screen readers.

In fact HTML5 semantic elements make the page structure more meaningful. Peterson, [46], suggests that an approach to have an accessible web page both for browsers that support HTML5 and other browsers or versions which don’t support it, is to create a web page using both HTML5 semantic elements and ARIA.

For example to have an accessible HTML header tag developer can combine HTML5 and ARIA:

```
<header role="banner"> ... </header>
```
In the example `<header>` is a HTML5 semantic element. An old browser does not know this element but it probably knows ARIA; so it renders it.

<table>
<thead>
<tr>
<th>HTML5</th>
<th>ARIA</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>header</td>
<td>role=&quot;banner&quot;</td>
<td>Starts a page or a section and can be designed as H1-H6.</td>
</tr>
<tr>
<td>nav</td>
<td>role=&quot;navigation&quot;</td>
<td>Can be used as site navigation, subnavigation, breadcrumbs, previous/next links</td>
</tr>
<tr>
<td>footer</td>
<td>role=&quot;contentinfo&quot;</td>
<td>Declares a page or section. It may contain author name, copyright info, etc.</td>
</tr>
<tr>
<td>aside</td>
<td>role=&quot;complementary&quot;</td>
<td>Main page content information that visually might be seen as a sidebar.</td>
</tr>
<tr>
<td>article</td>
<td>----</td>
<td>Independent item such as a blog post, article, etc. that can be moved around.</td>
</tr>
</tbody>
</table>

Table 1 Some of HTML5 elements and corresponding ARIA roles

As it is mentioned before, ARIA is integrated in HTML5. It means that HTML5 elements now have corresponding ARIA roles.

A web page structure contains different regions. A sighted user can see the page and work with different parts of it but screen readers help users by interpreting DOM. So it is not enough to have a structured visual interface. DOM needs to be structured well enough for the screen readers. HTML5 elements and ARIA with its landmark roles can be used to create this structure. Using HTML5 elements and ARIA landmark roles the developer can create semantic structure in DOM. [48]

Table 1 shows some of HTML5 elements and corresponding ARIA roles [46].

HTML5 tags such as `<audio>` and `<video>` increase accessibility too. There is a good reason for this: There is no need to use Flash or Java applets or any complicated code. These tags are just native tags and it means that they can be more accessible.

Also HTML5 has suggested some solutions to increase accessibility in forms too. Forms can be problematic for screen readers to interpret. They may be able to read labels and also read elements such as textfields as: Textfield. But they may not know which label is related to which textfield.

So using `for` attribute is a must to connect a label and a textfield.
For example:

```html
<label for="textfield-id">Username:</label>
<input name="textfield-name" type="text" id="textfield-id" />
```

This approach is supported for some other tags as well. For example now using HTML5, required fields can be natively supported. It means that developers can use `required` attribute. Also ARIA with `aria-required="true"` supports this.

So based on what is described in this chapter in order to have an accessible web page for a screen reader it is essential to have a clean DOM, logical page structure, descriptive labels and to use native HTML controls as much as possible. HTML5 and ARIA can help developers increase accessibility.

Best practices for document objects models include having logical sections in the model with grouped controls, labels for all images and controls and also using CSS instead of tables. In fact screen readers need structured documents in which keyboard interactions are supported, focus behavior is managed well and native HTML elements, ARIA are used as much as possible.
7. HTML5 Accessibility Solutions

In this chapter some of HTML5 accessibility solutions for some of the screen readers known problems are discussed. As it is mentioned in the previous chapters, what a screen reader needs to be able to interpret a website correctly is a good structured document object model. Document object model or DOM is what a developer creates to build a web page using HTML markup language. HTML5 as the latest version of HTML has provided some solutions, which make it easier to create a well-structured web page.

In this chapter some of the known problems for screen readers and HTML5 solutions for those problems based on accessibility principles are discussed. Of course to make it possible for screen readers to understand a web page correctly it is required to make them work better with the browsers that support HTML5. This paper focuses on the HTML5 markup language and compares it with the older versions of HTML; so more details about browser support for HTML5 and also developer role in creating an accessible website are out of scope of work.

7.1. Screen Readers Accessibility Problems

7.1.1. Skip Links and nav Element

In HTML4 lack of semantics leads to use more div tags in a web page development. In case of using screen readers, it is needed to be able to skip some part of the page and jump to another part as user wishes. Casario and his colleagues have written an interesting book "HTML5 Solutions: Essential Techniques for HTML5" [50] and they have included a chapter about some of the screen readers main problems and HTML5 accessible solutions. They explain that in HTML4, web page structure can be divided into headers, footers, navigational menu, body with page content, different columns, and so on. In HTML4, developer uses div tag to create these sections. For example to create the header section the developer may use:

<pre><div id="header"></div></pre>

Or for the footer section:

<pre><div id="footer"></div></pre>

And the body of the page can be something as below:

<pre>&lt;div id="body"&gt;
  &lt;div id="menu"&gt;&lt;/div&gt;
  &lt;div id="page content"&gt;&lt;/div&gt;
&lt;/div&gt;</pre>
And then CSS is required to add styles to each section. Also some Javascript code can be used to make it more interactive.

For a screen reader, which can only interpret document object model of a page, all of these used div tags are the same. So it cannot distinguish between them. WCAG guideline 2.4.1 says: "Bypass Blocks: A mechanism is available to bypass blocks of content that are repeated on multiple Web pages." [6]

It is possible that a user, who is using screen reader, doesn’t want to listen to the whole page content. It should be a way provided to this user to skip those parts that are not interesting to her. As it is mentioned before all of these div tags are the same for a screen reader. It cannot recognize which of them is navigation section and which is page content or side bar. In HTML4 way of work with lack of semantic tags, this can be done by skip links which are using a tag and having link anchors for different sections. In this way a user can jump to a desired section. To use skip links, developer needs to add link anchors for all different sections of the page and then user, using skip links can jump to each of them. There are other approaches to do this but discussing them is out of the scope of this thesis.

As it is mentioned in the previous chapters, the semantic elements of HTML5 make it much easier to structure a web page. Using HTML5 elements it is possible to create quick navigation functions and avoid creating skip links.

HTML5 provides a tag called nav that is there to specify navigation content. This tag allows the user to jump to the desired section of the page. This element can be skipped by screen readers or can be read when user wants it to be read. Here is a simple example of how to use a nav tag:

```html
<nav>
  <ul>
    <li><a href="#desiredsection"> Desired Section </a></li>
  </ul>
</nav>

<section id="desiredsection">
<p>
Some Content here!
</p>
</section>
```

When a suitable screen reader, which is created to work with HTML5 semantic elements, reaches the nav tag, it can skip it or read it based on what a user wants. The problem is that many of the current screen readers do not recognize the new semantic elements of HTML5. Screen reader providers need to work closely with W3C organization.

### 7.1.2. Tables

One of the most problematic situations using a screen reader to read a web page is when the web page contains tables as layout. Some developers use tables to create web page
layouts because sometimes it is easier but layouts made of tables are very confusing for users of assistive devices such as screen readers because the document object model of the web page in this case is very complicated and the screen readers get easily lost in this model. It is not possible to forget how much tables are important and useful in web pages. So it is essential to find a way to make them accessible and readable by screen readers. [50]

WCAG 2.0 guideline 1.1 says: “Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.” [6]

Tables, as non-text content, need to be understood and interpreted by screen readers in order to be read clearly for users. All information about what a table contains need to be provided for user agents. Tables need reasonable captions and understandable column names. Based on W3 organization, [51], HTML5 introduces a caption tag and also a summary attribute for table element which contains details about a table and what it contains. The summary attribute is not shown on page but it is available for screen readers to read it for their users.

```
<table summary="A summary about what table contains">
  <caption>
    A caption for this table
  </caption>
  ...
</table>
```

For a sighted user it is easy to see which data is in which column with what column name. For a blind user, it is screen reader that reads the table. Screen readers interpret data object model of the page and so they need more details to know which data belongs to which column. In HTML5 there is an attribute called header for td tag which contains the id of the header column of current column:

```
<thead>
  <tr>
    <th id="name">Name</th>
    <th id="birthdate">Birth Date</th>
  </tr>
  <tbody>
    <tr>
      <td headers="name">David</td>
      <td headers="birthdate">1982.03.22</td>
    </tr>
    <tr>
      <td headers="name">Sarah</td>
      <td headers="birthdate">1981.05.02</td>
    </tr>
  </tbody>
</thead>
```
In this example, screen reader, if it is adapted to work with HTML5, can understand that both David and Sarah belong to column name. In this case HTML5 has improved document object model to be interpreted better by screen readers than a document object model created using HTML4.

7.1.3. Forms

Forms are very important part of users and websites interactions. Forms contain different HTML elements such as input boxes, labels, and buttons and so on. HTML input tag has a type attribute that accepts different values. In HTML5 these values are increased by thirteen compared to HTML4. New values are tel, url, date, datetime, datetime-local, month, week, time, email, search, number, color and range. [52] When a screen reader reads the type attribute of an input box to its user, it will be clearer for the user to know what kind of data is expected. From the accessibility perspective it is essential to make it possible for user to navigate the form using keyboard. HTML5 provides accesskey attribute for label tags, which makes keyboard navigation possible. As it is mentioned in previous section about tables too, it is important to provide enough information for screen readers to be read for their users. For form element, a good explanation about what this form is for and what information is required and why it is a good step towards better accessibility. HTML5 introduces legend tag to provide a caption for a form element. Also HTML5 makes it possible to group form elements in several sets, which make the document object model more readable than a document object model created using HTML4.

7.1.4. ARIA Project

Based on Casario and his colleagues in “HTML5 solutions: Essential Techniques for HTML5”, [50], One of the known problems that screen readers were facing is dynamic page contents. At the beginning HTML web pages used to have only static contents. It means that when a page was loaded once, everything was sent to the client and no more communication between the client and server was needed. In this situation screen reader could read page content as it was. Later, client and server communication became more complicated. Using AJAX some parts of the page can keep sending and receiving data to the server while some other parts of the page are loaded once and not any more. The problem is that screen readers haven’t been able to know about the AJAX activity, which can happen. So they cannot update their users about these changes that occur on the page.

As it is discussed previously in this paper, W3C has a project called WAI-ARIA. [34] With ARIA it is possible to assign some properties to HTML elements that explain roles and functions associated with an element; so assistive technologies such as screen readers can be updated about the status of the element. It means that a screen reader can be informed of any change in the web page. In fact ARIA allows adding some properties to elements of a web page that can be interpreted by browsers and assistive technologies. ARIA can help assistive technologies in the following areas: [33]

- To detect hidden client-server interactions
- To navigate using keyboard in custom components
To add roles to the page structure

One of the most problematic situations for screen readers is when server sends new data to the client, user interface gets updated silently. For example with an implementation of AJAX a form element in a web page can be updated without any need to reload the page. ARIA has solved this problem by adding new set of properties to a region of a page that can detect the update and inform the screen reader. [50] These properties are:

aria-live: This element specifies that an element may be updated. It accepts two values. One is polite and the other one is assertive. When its value is polite, assistive technology notifies the user but doesn’t interrupt the current task. When the value is assertive, assistive technology updates the user immediately.

aria-atomic: Shows whether assistive technologies will present all or part of the changed region to their users.

area-relevant: Shows what changes are relevant within a region. It accepts additions, removals, text, addition text and all values. Its value is additions when elements are added to the document object model within the region. When elements or text are removed from the document object model within a region value is removals. When text is added to any element of the live region value is text. Using all is equal to the combination of all values "additions removals text" and additions text is equal to the combination of values "additions text".

area-busy: Shows whether an element is being updated. It accepts true or false as value.

So ARIA has provided a good way to notify assistive technologies about updates in regions of a document object model.

Also ARIA can help support navigation using keyboard. [32] As it is discussed in the previous chapters of this paper, mouse is not a good interaction device for people with sight disabilities. It is essential not to bind interactions with web pages to mouse events. ARIA has provided effective support for tabindex property. This makes it possible to create a navigation order for all visible elements on a web page.

7.2. How HTML5 Improves Accessibility

It is time to answer these questions: How does HTML5 improve accessibility in web development? Which principles and guidelines of WCAG 2.0 are addressed by HTML5? How HTML5 has made it easier for developers to obey these principles and guidelines?

In the previous chapters, this paper discussed accessibility and its principles and also different HTML5 features to increase the accessibility in design of web pages. Based on WCAG 2.0, [6], accessibility principles and guidelines are:

1. Perceivable
   It means that a website content including information and functional components must be clear and perceivable for users. For this principle there are four guidelines:
   1.1. Text alternative for non-text content.
   1.2. Alternatives for time-based media.
1.3. Possibility to present content in different ways.
1.4. Separated foreground and background.

2. Operable
All User interface component goals must be achievable and navigation in website must be clear, simple and meaningful. For this principle there are four guidelines:

2.1. Keyboard support.
2.2. Enough time to understand and work with content.
2.3. Preventing causes of seizures
2.4. Clear navigation.

3. Understandable
Content and functionality of user interface must be understandable. For this principle there are three guidelines:

3.1. Readable and understandable text.
3.2. Predictable appearance and operations.
3.3. Preventing users from making mistakes and providing ways to correct mistakes for them.

4. Robust
Website must be designed and developed robust because it needs to be interpreted reliably by user agents like assistive technologies not only for current ones but also in future.

Of course obeying provided guidelines is dependent to web developer's skills. It means that it is the developer responsibility to follow correct instruction and develop web pages based on that. HTML5 as a web development tool has provided some facilities to fulfil some of the mentioned requirements. This paper has discussed those facilities in different chapters. This section is presenting a summary of the results of the work.

WCAG 2.0 first principle is being perceivable. Guideline 1.1 is providing text alternative for non-text contents. HTML5 provides caption and summary tags for tables which developers can add details and summary about tables for screen reader. For images by figure and figcaption tags, developer can add a caption for an image. So screen readers now can know what the exact caption of an image is. Also in HTML5 there is a legend tag for forms which developers can add a caption for a form. It means that HTML5 has improved adding text alternatives in web development accessibility.

WCAG 2.0 second principle is being operable. Guideline 2.1 is providing keyboard support. HTML5 provides accesskey attribute for label tags which makes keyboard navigation possible.

HTML5 has also improved navigation in websites compared to HTML4. Guideline 2.4 declares that having a clear navigation is a must for a web page and a website. It is reasonable. As it is mentioned several times in this paper, screen readers only interpret document object model of a web page. They have no idea about the page appearance. What they need is to follow the page structure which is what document object model of the page presents to them. They need to navigate elements and different page sections only by following what they find in this model. HTML5 semantic elements help structure a web page much better than older versions of HTML. They help create a semantic document object model that a screen reader can interpret in a much easier and more correct way for its user. It is possible to distinguish between different sections of a web page. On the other hand ARIA roles help the document model become even more readable for screen readers. As it is mentioned in previous chapters, some web browsers still don't support HTML5 but ARIA has been there before HTML5 and so better support for ARIA is available by web browsers. HTML5 has ARIA as a part. So a
combination of HTML5 semantic elements and ARIA roles can lead to a very well-structured document object model.

As it is discussed in the previous section, HTML5 has introduced thirteen more values for input tag value attribute. These values are readable for screen readers so that when screen readers reach any input element in DOM, it can read the value attribute for its user and user can know what type of information is required. Also it is possible to group elements in a form. It makes the form more readable than before. It is possible to connect label elements and input boxes by for attribute. So screen reader can know which label belongs to which input box.

Also by introducing new table structure element, HTML5 has made it easier to interpret table elements in DOM. Adding header attribute to td tag helps in knowing which column is related to which header. It means that table navigation is much easier using HTML5 table element structure than using HTML4.

<table>
<thead>
<tr>
<th>HTML5 Solution</th>
<th>Accessibility Improvements</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML5 semantic tags</td>
<td>Better document object model</td>
</tr>
<tr>
<td></td>
<td>Better navigation possibility (Guideline 2.4)</td>
</tr>
<tr>
<td>Including ARIA in HTML5</td>
<td>ARIA roles help the document model to be more readable for screen readers</td>
</tr>
<tr>
<td></td>
<td>Better navigation possibility (Guideline 2.4)</td>
</tr>
<tr>
<td>HTML5 has added captions and summaries to many non-text elements such as tables, forms and figures</td>
<td>Helps screen readers to inform their users about the non-text page contents (Guideline 1.1)</td>
</tr>
<tr>
<td>HTML5 has introduced more values for input tag value attribute</td>
<td>when screen reader reads the value, user can may know what type of information is required</td>
</tr>
<tr>
<td>HTML5 has provided accesskey attribute for label tags</td>
<td>Better keyboard navigation (Guideline 2.1)</td>
</tr>
<tr>
<td>Grouping elements in a form</td>
<td>Better document object model</td>
</tr>
<tr>
<td>New table element structure</td>
<td>Better document object model</td>
</tr>
<tr>
<td>HTML5 has introduced video and audio tags</td>
<td>Makes the website more operable (Principle 2)</td>
</tr>
</tbody>
</table>

Table 2 HTML5 solutions to improve accessibility

As a conclusion HTML5 by improving web page document object model makes page navigation for screen readers easier than HTML4. It means that HTML5 improves accessibility of websites from navigation perspective too.

Also HTML5 video and audio elements make it possible to play video and audio in websites without any need to install external plugins. It can be counted as an improvement in
accessibility principle 2, being operable. It has been a problem to install external plugins even for users without sight disabilities. It is imaginable how hard it can be for a blind user.

Based on what is discussed in this section, HTML5 has improved accessibility of web pages at least by making them more perceivable and operable and specially more navigable compared to older versions of HTML. But it is important not to forget that as much as having good tool, such as HTML5, to develop a website is essential, having a right mindset about usability and accessibility is important too. Because without having correct mindset and will to make a website accessible, a rich developing tool cannot do much. Based on WCAG 2.0 guidelines, contents of a web page need to be readable and understandable. It is clear that HTML5, as a tool, cannot do anything about it. It is content manager responsibility to make sure about the content. It is web developer responsibility to think of different ways of presenting content to the users. Web developer is the one who has to find ways to prevent users from making mistakes or making it possible for them to correct their mistakes. At last but not least, it is web developer that needs to use all provided facilities by HTML5 to make the most use of it to create an accessible web page.
8. Conclusion and Future Work

8.1. Conclusion

This thesis has investigated how HTML5, as a web development tool, can improve accessibility in websites based on WCAG 2.0 accessibility principles compared to HTML4. It was necessary to concentrate on the main required factors which make a website accessible and then how HTML5 can help meet these requirements.

Among users with different kinds of disability, users with sight disabilities have been selected. Among different types of assistive technologies that they use, screen readers have been more interested in scope of the thesis.

As a result of literature reviews and case studies, the author of this paper believes that improvements in HTML which have concluded to introduce HTML5 and integration of WAI-ARIA into HTML5 has made these tools useful in order to have more accessible web pages. As it is discussed in detail in chapter 6 of this report, HTML5 semantic tags and including ARIA in HTML5 are important factors for a useful tool to develop more accessible websites. As it is discussed in section 6.2, HTML5 makes it possible to add captions and summaries to many non-text elements such as tables, forms and figures which are of a great help for screen readers to inform their users about the non-text page contents. Also HTML5 by providing accesskey attribute for label tags makes keyboard navigation possible. Good navigation possibility in a website is an essential factor for screen readers because they need to follow the page DOM structure. HTML5 semantic elements help structure a web page much better than the older versions of HTML. They help create a semantic document object model that a screen reader can interpret much easier and more correctly for its user. Also ARIA roles help the document model to be even more readable for screen readers.

More values for input tag value attribute have been introduced by HTML5. These values are readable for screen readers. So when a screen reader reads the value, the user can know what type of information is required. Also as it is discussed, grouping elements in a form makes the form more readable. By the new table element structure, HTML5 has made it easier to interpret table elements in DOM compared to HTML4.

As a conclusion HTML5 by improving web page document object model makes page navigation for screen readers easier. It means that HTML5 improves accessibility of websites from navigation perspective too.

Also in section 6.2 it is mentioned that HTML5 video and audio elements improve accessibility since there is no need to install external plugins.

Of course developer's knowledge about accessibility concept and tools such as HTML5 and her ability to implement a good semantic structure is very important. Designing and developing an accessible web page needs more than pure knowledge of software development. It also requires sufficient level of competence in creating semantic structure. Therefore close acquaintance with problems of disabled users along with psychological investigation besides technical knowledge can be efficient in this area.
It seems the advances in web technology and assistive technologies, in this case screen readers, lead to less difficulties in accessibility. As discussed in this report some of the HTML5 elements support accessibility as intended and only some of these elements are semantic. Still many browsers don't support HTML5 completely. At the same time older browsers are still used around the world by people. Also screen reader technology improves slower than web development technologies. It means that even an accessible web page sometimes cannot be interpreted well by a used screen reader.

8.2. Future work

Based on Zhang and Adipat in their paper "Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications." [53] Lack of studies in investigating accessibility of websites for other assistive technologies is apparent. Hence, as the future works on this thesis suggest more studies in this area. There are different kinds of assistive technologies which serve people with different types of disabilities. These assistive technologies have special requirements, which need to be implemented in web pages.

Another area which can be valuable to spend time is accessibility in mobile applications in order to interact sufficiently with assistive technologies. Mobile devices have smaller screen sizes and usually smaller memory capacities. Because of the Mobile context, one cannot expect the user to provide required power source available for her device. There are different operating systems and various characteristics for different devices, so the web based product needs to be adaptable to those who don’t use mouse or big keyboards for their mobile devices. Usually internet connection provided for mobile devices is not as rich as normal devices so it can be really hard or even impossible to download large files. Gafni in his paper "Usability Issues in Mobile-Wireless Information Systems" discusses more details. [54]

Based on Abascal and Civit in their paper "Mobile communication for people with disabilities and older people: New opportunities for autonomous life." [55] there are both benefits and difficulties in using mobile phones for people with disabilities. Mobile phones make it possible for these user group members to stay in contact with society and use maps and directions and also ask for help if they required any. So they can live more independently and do what they want with less help needed from others.

On the other hand, mobile devices can lead to some difficulties. These difficulties make it hard or impossible for some people to use these devices. For people with limited mobility, small buttons can be an obstacle to use the device. Or small screens make it really hard for people with a weak sight to read the information. Limited formats of providing feedback is another problem. People with some disabilities cannot make any benefits of some types of feedbacks. Also many assistive technologies that people with disabilities use on PCs still cannot be used on mobiles. Wobbrock in his paper "The future of mobile device research in HCI." [56] explains more about it.

Based on Newell and Gregor in their paper “Extra-Ordinary HumanMachine Interaction: What can be Learned from People with Disabilities?” [57] mobile device usage difficulties become more serious when we consider that these devices usually are used in a non-static environment. People use these tools in all different places. So many factors like noise or
weather or many other things can affect the accessibility. Cragun and Todd in their article "How do Mobile Apps Measure Up to Accessibility Standards?" [58], published article on IBM website February 2013, discuss more details on mobile accessibility factors.

Based on Kane and Bigham in their paper "Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques" [59] many people with disabilities have been successful to override some of the accessibility difficulties and currently are using these devices more happily and even some of them have learned or invented approaches to use the device as much as possible but still some functionalities are out of reach and inaccessible. It means a huge amount of work remains for accessibility and usability experts and also web developers to perform.

As a future work, this paper suggests more investigation on HTML5 and mobile accessibility. Because HTML5 has been a tool to develop mobile applications and mobile websites, doing research on mobile phone screen readers and also investigating accessibility factors which are important for these screen readers, to be able to read the mobile pages content for their users, can be a right area to spend time.
References


