Master

Localisation and web accessibility: a comparative study of the transfer of accessibility information through CAT tools

PACATI, Isotta

Abstract

By conducting a comparative study of two CAT tools, SDL Trados Studio 2017 and MemoQ v8.7, this research project aimed to examine the impact of these tools on the achievement of accessibility conformance during the web localisation process. Two different approaches were adopted to investigate this matter. The first stage comprised a descriptive approach, in which the researcher analysed the CAT tools' performance and determined whether they could support and transfer a selection of qualities related to accessibility embedded in an HTML5 code. The results indicated that the two systems supported and transferred the majority of these elements, but not all of them. The second stage of the research consisted of a tool evaluation carried out by ten novice web localisers. The goal was to measure the functional suitability of the CAT tools and analyse the influence of participants' knowledge of accessibility on the final target product.

Reference

UNIVERSITÉ
DE GENÈVE

Isotta Pacati

Localisation and web accessibility: a comparative study of the transfer of accessibility information through CAT tools

Directrice : Silvia Rodriguez Vázquez

Jurée : Lucía Morado Vázquez

Mémoire présenté à la Faculté de traduction et d'interprétation (Département Traitement Informatique Multilingue), pour l'obtention de la Maîtrise universitaire en traduction, mention Technologies de la traduction

Université de Genève

Année Académique 2019-2020

Janvier 2020

Déclaration attestant le caractère original du travail effectué

J'affirme avoir pris connaissance des documents d'information et de prévention du plagiat émis par l'Université de Genève et la Faculté de traduction et d'interprétation (notamment la Directive en matière de plagiat des étudiant-e-s, le Règlement d'études des Maîtrises universitaires en traduction et du Certificat complémentaire en traduction de la Faculté de traduction et d'interprétation ainsi que l'Aide-mémoire à l'intention des étudiants préparant un mémoire de Ma en traduction).

J'atteste que ce travail est le fruit d'un travail personnel et a été rédigé de manière autonome.

Je déclare que toutes les sources d'information utilisées sont citées de manière complète et précise, y compris les sources sur Internet.

Je suis conscient-e que le fait de ne pas citer une source ou de ne pas la citer correctement est constitutif de plagiat et que le plagiat est considéré comme une faute grave au sein de l'Université, passible de sanctions.

Au vu de ce qui précède, je déclare sur l'honneur que le présent travail est original.

Nom et prénom :

Isotta Pacati

Lieu / date / signature :

Vacallo, 21 Décembre 2019

Acknowledgments

I would like to express my very great appreciation to my supervisor Silvia Rodriguez Vázquez, who provided many interesting ideas at the beginning of this journey and helped me develop this thesis idea. Although I was lacking, thank you for always advising and guiding me until the very end. Thank you for your precious feedback and continuous support throughout the whole process. I would also like to thank Lucìa Morado Vázquez, who introduced me to the world of localisation and have sparked my interest in this field with her passion and enthusiasm.

Un grazie infinito alle mie compagne di Facoltà Flaminia, Sofia, Silvia, Paola, Raffaella e Benedetta, con le quali ho condiviso gli ultimi cinque anni. Grazie per essere state al mio fianco, per le ore passate chine sui libri, per le serate passate a ridere e per avermi aiutato e incoraggiato durante i momenti difficili.

Un ringraziamento speciale va alle mie amiche di una vita Alice S., Matilde e Alice M., che da sempre mi hanno incoraggiato e non mi hanno mai fatta sentire sola, nonostante la distanza. Grazie per le continue parole di supporto e per essere sempre con me, ovunque vada.

Ai miei genitori, a cui devo tutto. Grazie per tutti i vostri insegnamenti e per aver sempre creduto in me.

Abstract

In recent years, multilingual accessibility has become an issue of increasing relevance. By conducting a comparative study of two CAT tools, SDL Trados Studio 2017 and MemoQ v8.7, this research project aimed to examine the impact of these tools on the achievement of accessibility conformance during the web localisation process.

Two different approaches were adopted to investigate this matter. The first stage comprised a descriptive approach, namely a tool descriptive analysis, in which the researcher analysed the CAT tools' performance and determined whether they could support and transfer a selection of qualities related to accessibility embedded in an HTML5 code. The results indicated that the two systems supported and transferred the majority of these elements, but not all of them. In addition, the second stage of the research consisted of a tool evaluation carried out by ten novice web localisers. The goal was to measure the functional suitability of the CAT tools and analyse the influence of participants' knowledge of accessibility on the final target product. In this case, the findings suggested that CAT tools may have a positive impact on the achievement of localisation when localisers are not accessibility-savvy, as they offer useful additional information about the feature's context, and proved that localisers who know how to implement accessibility best practices can improve the overall accessibility conformance of the final product.

Keywords: *web localisation, web accessibility, HTML, CAT tools, EAGLES, CAT tool evaluation*

Contents

List of Figures

List of Tables

List of Abbreviations

ACA	Accessible Canada Act
API	Application programming interface
ARIA	Accessible Rich Internet Applications
AT	Assistive Technology
ATAG	Authoring Tool Accessibility Guidelines
CAT	Computer-Aided Translation
CRPD	Convention on the Rights of Persons with Disabilities
CSS	Cascading Style Sheet
DOM	Document Object Model
DPI	Disabled People's International
EAGLES	Expert Advisory Group on Language Engineering Standards
GALA	Globalisation and Localisation Association
GILT	Globalisation – Internationalisation – Localisation – Translation
GUI	Graphical user interface
HAMT	Human-assisted machine translation
HTML	HyperText Markup Language
ICT	Information and communication technologies
ISO	International Standard Organisation
ITS	Internationalisation Tag Set
ITU	International Telecommunication Union
LISA	Localisation Industry Standard Association
MAHT	Machine-assisted human translation
MLV	Multi-language vendor
MT	Machine translation
PDF	Portable Document Format

PHP   Personal Home Page

QA   Quality Assurance

SC   Success Criteria

SD   Standard Deviation

SMIL   Synchronised Multimedia Integration Language

TEnTs   Translation environment tools

TM   Translation Memory

TS   Translation Studies

UAAG   User Agent Accessibility Guidelines

W3C   World Wide Web Consortium

WAI   Web Accessibility Initiative

WCAG   Web Content Accessibility Guidelines

XHTML   eXtensible HyperText Markup Language

XLIFF   XML Localisation Interchange File Format

XML   eXtensible Markup Language

1 Introduction

This master's thesis focuses on the process of web localisation, more specifically on the localisation of accessibility information: it investigates whether Computer-Aided Translation (CAT) tools, used for the localisation of HTML files, can support and transfer all the features related to accessibility embedded in the code, and help localisers produce an accessible target file.

1.1 Research context

People's way of communicating and interacting has changed significantly since the advent of the Internet and the World Wide Web, together with the development of other digital technologies. In recent decades, information and communication technologies (ICT) have continued to grow exponentially (United Nations 2006, 169). According to the International Telecommunication Union (ITU), in 2018 more than half of the world population, around 51% (ITU 2018, 3) used the Internet, and it is expected to reach 75% by 2025 (ibid., 13). These figures highlight the central role that ICTs, principally the Web, have in our daily lives. As digital technologies affect nearly every aspect of life, every Internet user should access freely to the information provided online. However, this is not always the case, as shown in the Disability and Development Report, published by the United Nations' Department of Economic and Social Affairs: the percentage of people with disabilities which uses the Internet (19%) is lower compared to the percentage of people without disabilities (36%), as of 2011 (United Nations 2019, 173). Several reasons influence the lower use of the Internet, including accessibility-related issues. In other words, people with disabilities may face difficulties in accessing the information included in webpages if proper accessibility practices are not implemented.

ICTs can be a double-edged sword for disabled people. On the one hand, they "represent a powerful opportunity to improve quality of life, enhance inclusion and social engagement and make independent living possible" (ibid., 169), as users can have access to various online services, including public services, and e-learning materials, text-to-voice devices, etc. Therefore, digital technologies can help people with disabilities engage in society (ibid.). On the other hand, if digital technologies are not accessible, the opportunity of developing and maintaining their social life is taken away from them. Hence, we can affirm that "[u]nequal access to digital technologies brings about unequal participation in society" (Van Dijk 2005,

15). Moreover, unequal access creates a 'digital divide' (Goggin 2018, 1) between people with disabilities and people without disabilities.

1.1.1 Web accessibility

The general term 'accessibility' refers to the "quality of being easily reached, entered, or used by people who have a disability"[1]. Consequently, accessibility in the context of information and communication systems, in particular of websites, defines the extent by which the webpage is considered accessible, meaning barrier-free, to everyone: every user, including people with disabilities and the elderly, should have access to all the information available on the Internet (Yates 2005 in Qadri & Banday 2009, 2). In this specific case, then, it would be more accurate to talk about 'web accessibility', rather than just 'accessibility'.

The Web has facilitated access to information but, at the same time, it has aggravated the problem of exclusion of people with disabilities. This issue has pushed organisations, such as the World Wide Web Consortium (W3C), and governments to undertake measures that would ensure web accessibility, which resulted in several initiatives and enactment of legislations and laws (Qadri & Banday 2009, 3-4).

According to W3C, web accessibility "means that websites, tools, and technologies are designed and developed so that people with disabilities can use them" (W3C/WAI 2019), and it can be included in the concept of 'universal design'. The latter refers to the creation and development of products that can be used by everyone, within the widest range of situations (Henry, Abou-Zahra, & Brewer 2014, 1), including websites.

To propose a unified definition, Petrie, Savva and Power (2015) analysed 50 definitions of web accessibility. Their work resulted in the following definition:

> "all people, particularly disabled and older people, can use websites in a range
> of contexts of use, including mainstream and assistive technologies; to
> achieve this, websites need to be designed and developed to support usability
> across these contexts" (Petrie et al. 2015, 1)

---

[1] https://www.lexico.com/en/definition/accessibility [11.06.2019]

According to the authors, having a unified definition that includes all the six core concepts[2] found in the other definitions avoids running the risk of creating a product, in our case a website, which is not accessible (ibid.).

As the Web is becoming more and more complex, web developers and web designers should know how to make their webpages accessible. To do so, in 1997, W3C launched the Web Accessibility Initiative (WAI), which "provides an international forum for collaboration between industry, disability organizations, accessibility researchers, government, and other interested in Web accessibility" (W3C/WAI 2019a). Among all its tasks, WAI develops a set of guidelines used by governments and organisations as international standards for web accessibility (Qadri & Banday 2009, 2): the Web Content Accessibility Guidelines (WCAG) (Kirkpatrick et al. 2018), which we will illustrate in Chapter 2.

1.1.2 Disability

The concept of disability is complicated and multidimensional, and many attempts have been made to define it from numerous perspectives (Altman 2001, 97-98). As stated in the United Nations' Convention on the Rights of Persons with Disabilities (CRPD), people with disabilities "include those who have long-term physical, mental, intellectual or sensory impairments which in interaction with various barriers may hinder their full and effective participation in society on an equal basis with others" (United Nations 2006, 4). This definition introduces two important concepts for our research: impairment and barrier.

The social model of disability redefined the concept by making a distinction between 'disability' and 'impairment': "[d]isability is something imposed on top of our impairments, by the way we are unnecessarily isolated and excluded from full participation in society." (UPIAS 1975 in Shakespeare, 2010, 198). Therefore, we can state that impairment is a term that refers to a physical or mental limitation, while disability refers to the social exclusion (ibid.): by providing accessibility solutions, the individual has an impairment but can still take part in social activities. This argument also applies to web accessibility. For instance, if a blind person navigates a website in which the right accessibility practices are implemented, said person will still be visually impaired but will be able to access the information.

Moreover, the Disabled People's International (DPI) included the concept of 'barrier' in its definition of disability, which is described as "the loss or limitation of opportunities to take part

---

[2] Six core concepts: groups of users, characteristics, needs of users; what users should be able to do; technologies used; characteristics of the website; design and development of the website; characteristics of the situations of use.

in the normal life of the community on an equal level with others due to physical and social barriers" (Oliver 1996, 33). In the context of web accessibility, we can talk about digital barriers that prevent people with disabilities from accessing the information, namely obstacles that can be avoided if sufficient practices are implemented. Among the numerous barriers that exist in the digital world, we find audio content without captions or transcript that are not accessible to people with auditory disabilities, complex navigation mechanisms and layout that can be confusing for people with cognitive, learning and neurological disabilities, websites that do not provide keyboard support for people with physical impairments, and non-text content that does not have an equivalent text alternative for people with visual impairments (W3C/WAI 2017).

The difficulty in defining the concept of disability is also due to the complexity of its nature. It is fundamental to recognise that disability is an evolving concept and to recognise the diversity of people with disabilities (United Nations 2006, 1-2). There is a variety of impairments that need to be considered when elaborating standards and practices related to accessibility. For instance, visual, auditory, and motor disabilities, together with cognitive problems, were targeted by WAI when drafting the first version of the WCAG (Chisholm, Vanderheiden & Jacobs 1999).

1.1.3 Assistive technologies

As we have introduced in the previous section, physical limitations and mental impairments can prevent people with disabilities to use digital products, in our case websites, which are not designed according to their needs. However, there are many ways for disabled people to navigate the Web, depending on their abilities (W3C/WAI 2017a). One option to access the information online is to provide the individual with a tool, such as assistive technology (AT) (Vanderheiden 1998, 30).

In general, AT refers to "any item, piece of equipment, or product, whether it is acquired commercially, modified, or customized, that is used to increase, maintain, or improve the functional capabilities of individuals with disabilities" (WHO 2011, 101). ATs have the advantage to be adapted to the needs and abilities of the individual, in addition to increasing independence and participation (Vanderheiden 1998; WHO 2011). In this research, we are interested in ATs that enable individuals with disabilities to use the Web.

The authors of WCAG defined AT as a "hardware and/or software that acts as a user agent, or along with a mainstream user agent, to provide functionality to meet the requirements of users with disabilities that go beyond those offered by mainstream user agents" (Kirkpatrick et al.

2018), which provides functionalities such as alternative presentations, alternative input methods, additional navigation or orientation mechanisms, and content transformation. Useful ATs to access a graphical user interface (GUI) are visual reading assistants, such as screen magnifiers, that allow the user to change text font, size, spacing, and other components to improve the visual readability; screen readers, which allow blind people to access information for non-text content; text-to-speech software; alternative keyboards, for instance, head pointers or single switches, and alternative pointing devices, which simulate respectively the keyboard and the mouse pointing and button activations (ibid.).

1.2 Motivation

As stated at the beginning of this thesis, the Internet has become a channel to convey information about numerous topics, including health and education, and has become a place where people can have access to many public services. "Accessing general information online enables people with disabilities to overcome any potential physical, communication and transport barriers in accessing other sources of information" (WHO 2011, 183-184); therefore, ICT equipment and services should be designed to benefit not only the wider population but also people with impairments (ibid.). However, only in Europe, less than 10% of websites are accessible, and 5% of the European population does not use the Internet due to an impairment (European Commission 2019, 1).

A multi-sectoral and multi-stakeholder approach can be an option to guarantee accessibility: "[g]overnments, industry and end-users all have a role in increasing accessibility" (WHO 2011, 186). As suggested by WHO, the industry itself plays an important role in ensuring access to accessible information. It is fundamental for all actors involved in the web cycle to gain a certain level of expertise to guarantee a sufficient degree of accessibility (Rodríguez Vázquez 2013, 384). Among these actors, we can also include web localisers. Localisation, as we will discuss in greater detail in Chapter 2, can be described as the process of adapting a digital product to a language and culture. With the term web localisation, we define a specific type of localisation that consists in the adaptation of the content included in a website (see Section 2.1.1).

A study aiming to define localisers' role during accessibility assessment tasks showed that localisers' participation would contribute to the achievement of web accessibility, in particular for text and graphic content in the context of multilingual websites (ibid, 387). There are many

aspects that a localiser should consider, as highlighted by Torres del Rey and Morado Vázquez (2019):

> "If content is […] linguistically, culturally, semantically, pragmatically and technically accessible for source locale users, those synergies must be reconstructed in a different locale, with different expectations and experiences among communities of people with disabilities. It is therefore logical to assume that success in localizing accessible content depends on how that content is transferred and, if necessary, adapted and transformed linguistically, culturally and technically in relation to its surrounding context, and linguistic, cultural, semantic, pragmatic and technical features, and according to target users and use environment." (Torres del Rey & Morado Vázquez 2019, 6)

In addition to the above, we also need to consider the influence that the tools used during the web localisation process, such as CAT tools, can have on the achievement of web accessibility. Do they support the localiser's effort in ensuring accessibility? Do they have a positive or negative impact? With these questions in mind, through our work, we aim at contributing to the current research in the field by analysing the impact that CAT tools have on the degree of accessibility achieved in the final target product in the context of localisation.

1.3 Research questions and hypotheses

As introduced at the beginning of this master's thesis, this work will focus on one of the steps (use of CAT tools) of the process of web localisation (defined in Chapter 2). In the previous sections, we reviewed how people with disabilities use the Web and highlighted the importance of web accessibility. In this section, we outline the main objective of this research, the research questions and the hypotheses on which we based the design of our study.

Web accessibility does not only concern the design and creation of a web-based GUI, but it should also be taken into account during the localisation process. One could assume that accessible source files, if localised correctly, will produce accessible target files. On the contrary, if the information and techniques supporting accessibility are ignored or not transferred correctly, the degree of accessibility of the target file will be inferior in comparison to the source, and, consequently, people with disabilities being part of the target audience will not be able to access the information properly. In the particular context of this thesis, we are interested in what we call 'accessibility features'. By this term, we understand the

characteristics that help achieve accessibility, namely elements (coding elements, attributes, text units, etc.) that can be embedded in the code to ensure accessibility.

Computer-Aided Translation (CAT) tools are often employed by localisers to translate HTML-based files. These tools usually support this type of files: they identify and isolate the translatable or localisable information and protect the code that in principle should not be modified. As certain accessibility features are often embedded in the code, there is the risk that they might be overlooked or even not supported by the CAT tool in use. Taking into account this initial premise, our research aims at exploring the following questions:

> **RQ1.** Can CAT tools support and therefore transfer all relevant accessibility features when processing an HTML5 file?

And therefore,

> **RQ2.** Is the resulting target file accessible?

The relevant literature (Chapter 2), more specifically researches related to multilingual web accessibility and localisation, together with researches on HTML-format file support in CAT tools, led us to formulate the following hypothesis:

> **H1.** CAT tools do not support and transfer all the accessibility features.

To answer the second question and examine the functional correctness of the tools studied, we conducted an experiment, which we will describe in Chapter 3. The experiment sought to confirm or reject the following hypotheses:

> **H2.** The functional completeness of the tool used has an impact on the final degree of the accessibility achieved.

> **H3.** The functional appropriateness of the tool used has an impact on the final degree of accessibility achieved.

> **H4.** The participants' level of knowledge of accessibility has an impact on the final degree of accessibility achieved.

1.4 Methods

To answer the research questions, we adopted two different approaches. To determine if CAT tools can support and transfer the information features concerning accessibility, we adopted a descriptive approach and performed a tool descriptive analysis of two CAT tools: SDL Trados Studio 2017 and MemoQ v8.7. The researcher examined the tools' features and functionalities

when processing an HTML5 file to determine if the accessibility features selected for our study (defined in Chapter 3) were supported correctly and had the potential to be transferred to the target document accordingly.

As we introduced in the previous section, to answer the second question we conducted an experiment with novice web localisers. Firstly, participants were asked to answer a preliminary questionnaire to determine their background, their level of knowledge of accessibility and localisation, and their experience with CAT tools. This questionnaire helped define the hypotheses and the test that followed. Secondly, participants were asked to localise an HTML5 file with both CAT tools and answer a post-evaluation questionnaire. The design of the study was based on the seven steps proposed by the Expert Advisory Group on Language Engineering Standards (EAGLES) (EAGLES 1999), which will be explained in Chapter 3.

1.5 Structure of the thesis

In this introductory chapter, we illustrated the research context, focusing on the definition of disability and the importance of web accessibility, the questions and hypotheses, and two methods used in this thesis. The following chapter (Chapter 2) provides an overview of the fundamental concepts of our research, namely localisation, accessibility, and HTML, related works, and a comparison of theories. Chapter 3 includes the methodology established to carry out both the tool descriptive analysis and the user evaluation. Chapter 4 describes the data collected through the first stage of our research, while Chapter 5 illustrates the results of the experiment carried out by participants and discusses the findings of our work. Finally, in Chapter 6, we draw the conclusions of our work and set new avenues for future research in this area.

2 Literature Review

In this chapter, we analyse the fundamental concepts of this master's thesis: localisation, accessibility, and CAT tools. In the first section (2.1), we illustrate how the industry and Translation Studies (TS) scholars define localisation, and describe in more details the specific type of localisation which will be the object of our research, namely website localisation, as well as the most popular markup language used to develop web content, HTML. Section 2.2 deals with web accessibility, specifically the set of guidelines and techniques that are usually employed to make a website accessible. In Section 2.3, we examine the relationship between the topics reviewed in the two previous sections, we define the concept of 'accessibility features' and provide an overview of the most relevant accessibility success criteria for web localisation. Finally, in Section 2.4, we review the main functionalities of Computer-Aided Translation (CAT) tools, prior studies on CAT tools, localisation and accessibility, and the evaluation methods that exist to assess this type of software.

2.1 Localisation

In this section, we introduce the first fundamental notion of this thesis: localisation. Before analysing how Translation Studies (TS) scholars and the industry define this concept, we will briefly retrace its history: how it all started and how it developed through the decades. People in the industry started talking about localisation in the late 1970s and early 1980s when personal computing and software became popular among people who did not possess programming skills. Economic reasons were central for the evolution of localisation, as companies such as Microsoft and Oracle wanted to export their products, already popular in the U.S., in other countries like Japan and the so-called FIGS countries (France, Italy, Germany, and Spain) (Jiménez-Crespo 2013, 8).

At first, developers hired linguists to translate textual strings and establish translation practices. However, they soon realised that separating the software from the translation only posed technical challenges, as it required translators to possess a basic knowledge of programming (ibid., 9). This factor and the formation and development of multi-language vendors (MLVs) in the mid-1980s, together with an extended outsourcing model which became popular in the 1990s, are among the factors that started the transition from translation to localisation (Esselink 2000, 5-6). Nevertheless, the constant demand for new target languages led to a reshaping in the industry in the 2000s. At present, the localisation industry has become more and more

complex, since new technological developments have been introduced (Jiménez-Crespo 2013, 9).

But how do experts in the industry and TS scholars define this concept? The term comes from the notion of 'locale', defined by the ISO standard 17100 (ISO 2015) as the "set of characteristics, information or convention specific to the linguistic, cultural, technical, and geographical convention of a target audience" (ibid.). Likewise, Pym (2004) defines it as the ensemble of a particular variety of a language and local convention (such as currency, dates, etc.) (ibid., 2).

Localisation can be defined by maintaining this notion of 'locale', as we can see in the definition given by the now-defunct Localisation Industry Standard Association (LISA): "[l]ocalization involves taking a product and making it linguistically and culturally appropriate to the target locale (country/region and language) where it will be used and sold" (LISA 2003, 13 in Jiménez-Crespo 2013, 13). However, in 2007, LISA itself expanded its own definition to include a large range of service and adopted the notion of 'market' instead of 'locale': "[l]ocalization is the process of modifying products or services to account for differences in distinct markets" (ibid.). This second definition highlights the relevance of economic reasons that lie behind the localisation industry, as it is part of a broader and global cycle.

Similarly, the Globalisation and Localisation Association (GALA) also adopted the notion of 'locale' in its definition and described localisation as "the process of adapting a product or content to a specific locale or market" (GALA 2015). In addition, GALA listed all the elements that are included, besides translation, in the localisation process: "adapting graphics to target markets, modifying content to suit the tastes and consumption habits of other markets, adapting design and layout to fit translated text, converting to local requirements (such as currencies and units of measure), using proper local formats for dates, addresses, and phone numbers, and addressing local regulations and legal requirements" (ibid.). Through this definition, we realise the complexity of the whole process: localisation does not consist only of the adaptation of the original text, but there are numerous aspects that localisers should consider. Among these, we could also add what we define as 'accessibility features', that we will further explain later in this chapter.

From a TS perspective, we can identify two major trends that are followed by scholars. On one hand, localisation is considered as a 'translation-related phenomena', as a mere translation modality shaped by technological and project-based features. On the other hand, the second trend focuses on the description of industrial practices through a professional approach

(Jiménez-Crespo 2013, 17). For instance, concerning the latter trend, Dunne (2006) gave the following definition, which includes the concept of locale and adds the notion of non-text content:

> "[t]he processes by which digital content and products developed in one locale (defined in terms of geographical area, language and culture) are adapted for sale and use in another locale. Localisation involves: (a) translation of textual content into the language and textual conventions of the target language, (b) adaptation of non-textual content (from colors, icons and bitmaps, to packaging, form factors, etc.) as well as input, output and delivery mechanism to take into account the cultural, technical and regulatory requirements of that locale (ibid., 4)."

As we previously mentioned, to understand the concept of localisation, we need to consider it as part of a broader and global cycle, "a much wider complex of interrelated processes known as GILT" (Jiménez-Crespo 2013, 24): Globalisation – Internationalisation – Localisation and Translation cycle.

In the context of the language industry, with the term globalisation, we refer to the broader processes in the cycle necessary to offer products and activities on a global scale. This concept not only includes language-related aspects such as multilingual communication but also practices related to commerce and trade policies (GALA 2015a).

Likewise, internationalisation refers to the stage that precedes the development of a digital product (Jiménez-Crespo 2013, 25). This process can facilitate the localisation task, as it ensures that a product "can be adapted to various languages and regions without requiring changes to the source code" (GALA 2015b). In the context of our research, the internationalisation of a website allows users from different locales to access equally to information. This process is usually carried out during the product development cycle prior to the localisation process (Esselink 2000, 2-3). Since web localisation is at the centre of website internationalisation, we refer to the publication of a multilingual website as 'website globalisation', which includes the internationalisation of the back-end software, the design of a multilingual architecture, and the localisation of the site's content (ibid., 4-6). In this research, we will focus on the latter to illustrate how localisation practitioners can obtain a multilingual website.

The following step concerns localisation, which we have just defined above. In the GILT cycle, this stage includes preparation, management, engineering and quality assurance (QA). Therefore, the fourth step, the translation process, can be described as the transfer of textual

material by the translator-localiser (Jiménez-Crespo 2013, 26). In this research, we will focus on the third and fourth stages, namely the actual localisation and translation processes, in which the product is adapted to the target locale.

Last, it is worth mentioning here that localisation, as a general term, refers to numerous types of localisation practices, such as software, videogame, application and website localisation. In this thesis, we will focus on the latter, which we will expand on in the following section.

2.1.1 Web localisation

Website localisation, or just web localisation, emerged in the latest 1990s and it is often included in more general definitions of localisation. For instance, Esselink (2000) describes the general notion as the "translation and adaptation of a software or a web product" (ibid., 1). Therefore, we can state that web localisation is the process of adapting the content of a website to "make it accessible, usable, and culturally suitable to a target audience" (Sandrini 2008, 9). In this specific case, the author highlights how the content should be adapted to a specific audience, in all its diversity. As we will see in Section 2.2 and later in Chapter 3, the audience comprises diversified individuals with varying abilities and, consequently, the product should be adapted according to everyone's needs, including people with disabilities'.

Moreover, in the attempt of defining localisation in a more general way, Jiménez-Crespo (2013) described web localisation as "a process by which interactive digital texts are modified to be used in different linguistic and sociocultural contexts, guided by expectations of the target audience and the specifications and degree requested by initiators" (ibid., 20). Here, specifications and the degree requested by initiators could also refer to the information concerning accessibility that should be respected and transferred correctly to obtain an accessible product in the target locale.

Before progressing further with our literature review, it is fundamental to understand the type of content that the localiser is normally asked to adapt and translate, as localisation does not only concern language issues (Pym 2011, 5). Sandrini (2008) divides the website content, composed of digital assets, in six categories: common content, multimedia assets, applications-bound assets, transactional assets, and community assets (ibid.). The localisable content can be included in those six categories and concerns mainly text units, which can be found in the title, descriptions, keywords, menus, hyperlinks, description of non-text content (such as images, videos, graphics, etc.), and in audio-visual files (Pym 2011, 2).

Website content is subject to a 'content life cycle', "which describes the usability of the information from the time of creation to publication and finally archiving"(Sandrini 2008, 10). The information is stored in web documents, using markup languages such as XML (eXtensible Markup Language) or HTML (HyperText Markup Language) (ibid.). In the following section, we define the latter to have a clear understanding of the markup language we will work with.

2.1.2 HyperText Markup Language

The acronym HTML stands for HyperText Markup Language and refers to the core language of the World Wide Web (W3C 2018). In 1989, when Tim Berners-Lee invented the Web, he used HTML as its publishing language with the idea of gathering information from researchers all around the world and link all their studies to each other, to obtain cross-references from one research to another. In simpler words, HTML is the text format of websites (Raggett et al. 1998). In its first ten years, until 1998, this markup language kept developing, resulting in the HTML 4.0 version. As people in the industry started focusing on different types of languages to provide web content, HTML was put aside. However, as new technologies such as XForms were introduced, the industry started working on a new version, introduced in 2014: HTML 5.0 (W3C 2018). The latest version, HTML 5.3, was published on the 18th October 2018[3].

HTML5 is based on four design principles: compatibility, utility, interoperability, and universal design. The latter concerns directly our research, as it includes accessibility among its three concepts (the other two being media independence and support for all world languages). According to this concept, design features should be accessible to everyone regardless of their abilities, particularly to people with disabilities (Lawson & Sharp 2010, 3-5; Lubbers, Albers & Salim 2010, xiii; W3C 2007).

Moreover, HTML5 specifications include what has previously been specified in HTML 4.1, XHTML[4] 1.1 and DOM[5] Level 2 HTML (W3C, 2018) and "it has moved HTML from being a relatively simple document markup language to being a sophisticated platform for web applications with a host of new, rich application programming interfaces (APIs)" (O Connor 2012, 1). Therefore, new features and new elements (see Appendix A) were introduced in this

---

[3] https://www.w3.org/TR/html53/introduction.html#introduction [15.6.2019]

[4] **Extensible HyperText Markup Language**: an almost identical but stricter HTML defined as an XML (eXtensible Markup Language) application.

[5] **Document Object Model**: it defines HTML elements as objects and their properties, methods, and events. It is also an API for JavaScript.

latest version, which often presents many challenges to users, and especially to people with disabilities (ibid.).

This new version has a host of elements that include a broad range of new functions. First, developers introduced new semantics: elements, attributes, and attribute values have specific meanings and they are used only for their intended purpose (O Connor 2012, 128; W3C 2018a). Thus, new semantic markup is used to describe the element's content (Lubbers et al. 2010, 10). In the context of our research, we deem it important to emphasise that semantic information is essential to ensure accessibility, as assistive technologies use it to provide information to present the document and to provide additional functionalities (W3C 2018a).

We can use semantic information to structure the document. HTML5 provides many new sectioning elements, such as `<header>` (which includes the header's content, at the top of the page) and `<section>` (which include a section's content in the document's body), as we illustrated in Figure 2.1.
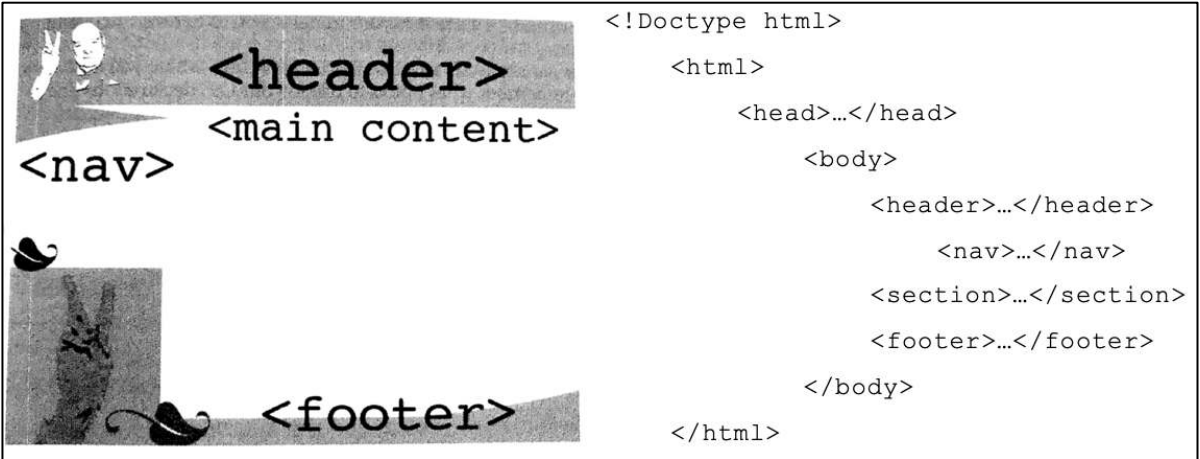
Figure 2.1: Basic structure of an HTML document (adapted from O Connor 2012, 167; Lubbers et al. 2010, 11-12).

Second, every element includes information about the category it belongs to, the context it is used in, the 'content model' that defines the element's content, and the DOM interface implemented (O Connor 2012, 142-3). Concerning the category the element belongs to, we can find different types of content; however, one element can have different characteristics, and for this reason, it can fall under several categories. As illustrated in Table 2.1, content can be divided into the following categories: metadata, flow, sectioning, heading, phrasing, embedded, and interactive (Lubbers et al. 2010, 10; O Connor 2012, 143-52; W3C 2018a).

14

Content model	Description
Metadata content	This kind of content outlines the presentation of the page, the behaviour of the content, and the relationships with other documents (W3C 2018a).
Flow content	It includes all the main elements used in the body of a document (ibid.).
Sectioning content	It defines new sections that can be grouped together and defines the scope of a certain part of the document, such as headings and footers (O Connor 2012, 147; W3C 2018a).
Heading content	This kind of content helps determine the structure of the page and defines the header of a section (O Connor 2012, 147; W3C 2018a).
Phrasing content	"This is the main body of text in a document and the inline elements used to mark up the content" (O Connor 2012, 147). In the context of content models, the term 'text' defines nothing or text nodes, which consist of Unicode characters[6] (W3C 2018a).
Embedded content	This kind of content allows the import of another source in the HTML document (O Connor 2012, 150).
Interactive content	It defines the "content that is specifically intended for user interaction" (W3C 2018a).

Table 2.1: Content model categories.

Last, among all the new HTML5's elements, we should talk about one crucial aspect in the context of accessibility and our research: WAI-ARIA. WAI-ARIA (Accessible Rich Internet Applications) is a "technical specification that provides a framework to improve the accessibility and interoperability of web content and applications" (W3C/WAI 2017b). This specification is essential to ensure accessibility, as it allows to include more information than the current HTML5 specification can provide (O Connor 2012, 135). An ARIA element is usually added to the native HTML element (ibid., 140) and, except cases where there are restrictions, web developers can use the ARIA role or `aria-*` attributes (W3C 2018a). For instance, when an HTML element is followed by the attribute `hidden`, the author can

---

[6] The **Unicode Standard** ist he universal character encoding standard for written characters and text: https://www.unicode.org/versions/Unicode12.0.0/ch01.pdf [17.6.2019]

introduce the ARIA semantics 'aria-hidden' and set it to 'true' (O Connor 2012, 139). In this way, a screen reader will ignore the element and avoid to read it to the user.

2.2 Web accessibility

As we have seen in Chapter 1, web accessibility consists of developing websites and technologies that are designed for everyone, in particular people with disabilities. In the following section, we briefly examine the WCAG guidelines, as they provide the basis for the selection of the research material and the selection of a set of requirements.

2.2.1 Web Content Accessibility Guidelines

The Web Content Accessibility Guidelines (WCAG), published for the first time in 1999, are a set of recommendations for making the content of the Web accessible. It was developed through the cooperation of W3C (specifically of the Accessibility Guidelines Working Group), individuals, and organisations around the world. The latest version of this document, WCAG 2.1 (Kirkpatrick Andrew et al. 2018), was published on the 5th June 2018. The set of guidelines is divided into four principles (perceivable, operable, understandable, and robust); thirteen guidelines, which provide basic goals to authors in order to make accessible content; and 77 success criteria (SC), divided into three levels of conformance (A, AA, AAA) and written as testable statements that are not technology-specific, and followed by a wide variety of techniques (ibid.), 597 in total.

Techniques offer specific guidance for web developers and evaluators to meet the WCAG success criteria and make their websites accessible and are divided into three categories: sufficient techniques, advisory techniques, and failures. The first type includes the techniques that should be implemented, while the second is "suggested to improve accessibility" (W3C/WAI 2019b), but they may not be sufficient and web browsers and/or assistive technologies may not support them. On the contrary, failures define the cause of accessibility barriers and, therefore, they should be avoided (ibid.).

Moreover, techniques can also be general or technology-specific. In the following table (Table 2.2), we briefly describe the different types of techniques.

Techniques	Description
General	In this category, we find, as the name says, general techniques. It is important to clarify that technology-specific techniques do not replace the general ones, but they should be considered at the same level (W3C/WAI 2019c).
ARIA	As we have seen in section 2.1.2, WAI-ARIA helps to add information in HTML elements. ARIA techniques provide ways to implement these elements and make them accessible, with the purpose of describing the content of the web page (ibid.).
Client-Side Script	This kind of techniques includes all the scripts that the client, or the user, can see. In this way, the user can modify and manipulate the content of the page according to their needs (ibid.).
CSS	CSS (Cascading Style Sheet) is used to modify the appearance of a web page and can be used to resolve issues related to contrasts, font size, and media support (ibid.).
Flash	This type of technique concerns Adobe Flash Player (a cross-platform browser plug-in), which allows adding videos, vector-based graphics, and other elements to the website. Therefore, it is crucial for authors to follow these techniques so that this kind of content is still accessible to people with disabilities.
HTML	These techniques give recommendations concerning HTML and XHTML, namely how to ensure accessibility by using the appropriate HTML elements and attributes (ibid.).
PDF	PDF (Portable Document Format) is a file-format that represents independent documents. This type of techniques helps authors describe the logical order of content and present the content of each part (ibid.)
Server-Side Script	Contrary to Client-Side Script, this type of technique includes all the scripts that the server itself should manipulate to ensure accessibility (ibid.).
Silverlight	Microsoft Silverlight is a tool used for creating interactive user experiences and mobile applications. Built-in support is provided by Silverlight for accessibility (ibid.).

SMIL	SMIL (Synchronized Multimedia Integration Language) is a markup language used for multimedia elements, such as videos and audios. It is important for the author to add descriptions and captions when dealing with this type of content (ibid.).
Plain-Text	This technique concerns the text format for paragraphs, lists, and headings (ibid.).

Table 2.2: WCAG techniques.

However, to ensure accessibility, we also need to consider several components of web development, namely content, web browsers, media players, assistive technology, users, developers, authoring tools, and evaluation tools (W3C/WAI 2018a). To address all these aspects, WAI published two additional documents, which together with the WCAG constitute the three-part approach of W3C: the Authoring Tool Accessibility Guidelines (ATAG)[7] and the User Agent Accessibility Guidelines (UAAG) [8].

2.3 Web accessibility and web localisation

In Section 1.1.1 and Section 2.1.1, we defined both web accessibility and web localisation, respectively. But how are these two concepts related to each other? As we introduced in the first chapter, localisers can contribute to the achievement of web accessibility (Rodríguez Vázquez 2013). In this section, we will analyse some articles and studies by different scholars, who examined the relationship between web accessibility and localisation.

In an informational article, Ó Broin (2004) illustrates how he initially believed that there might be a conflict between accessibility requirements and localisation. However, he then realised that localisation could be, in fact, a form of accessibility, because through localisation we deliver equivalent information and, consequently, make the content accessible to different audiences. In addition, he also listed several areas that concern both localisation and accessibility: language detection, a clear understanding of texts, text externalisation from graphics and multimedia, abbreviations and acronyms, tabular data, and separation of content from presentation. As we will see in future sections, these elements appear often on accessible websites and, according to the author, localisers should be aware of these specific accessibility features, as they will have positive implications on the general localisation process (ibid.).

---

[7] https://www.w3.org/WAI/standards-guidelines/atag/ [11.6.2019]

[8] https://www.w3.org/WAI/standards-guidelines/uaag/ [11.6.2019]

In another educational article titled "Fostering accessibility through localisation", Rodríguez Vázquez and Torres Del Rey (2014) embrace this idea of localisation as a form of accessibility and highlight how "accessibility achievements in the monolingual site could be undone" (ibid., 35) if localisers do not transfer the same level of accessibility achieved in the original code. The authors state that the final target product should be as accessible as the original one and that localisers could contribute to web accessibility audits by working with the original development team of the product, sharing knowledge and collaborating among the actors of the localisation workflow (ibid., 36-37).

In a subsequent article focusing on accessibility as a key factor in localisers education, the same authors (2016) argue that "[t]raining localisers from the (disad)vantage point of accessibility allows teachers and students to become aware of what the product does, means, can be used for; who, and how, its beneficiaries and users would be; and, most importantly, how the above is achieved and can be achieved (or adapted for functional diversity) in different locales or cultural contexts" (Torres del Rey & Rodríguez Vázquez 2016, 975). Consequently, when a localiser is aware of how accessibility best practices are implemented, the degree of accessibility achieved in the target product could be superior compared to the source. This result was demonstrated in a study carried out by Rodríguez Vázquez (2016), who focused on images localisation: accessibility can be achieved not only through the translation of text alternatives, but also through the assessment of their suitability (ibid.). We consider this aspect significant for our research, as we will also try to determine whether the level of localisers knowledge of accessibility can have an impact on the final degree of accessibility.

Nevertheless, from an industry perspective, a qualitative study carried out by Rodríguez Vázquez and O'Brien (2017), in which fifteen representatives of six language service providers were interviewed, suggested that there is still a lack of awareness on the importance of accessibility and, consequently, it is not automatically taken into account during the localisation process (ibid.).

2.3.1 Multilingual web accessibility

The element of multilingualism is barely contemplated in studies and official documents that concern accessibility. In 2016, Rodríguez Vázquez argued that "at present there is no standardised procedure for assessing the accessibility level of multilingual website" (Rodríguez Vázquez 2016, viii).

Since then, this aspect has been further studied in several theses. Casalegno (2018) investigated the impact of partial localisation strategies on the web navigation experience of screen reader users when browsing multilingual websites. Her study demonstrated that users encountered more difficulties in navigating localised versions of a multilingual website compared to the original version, due to language-related issues, technical issues (which concern the interaction between websites and screen readers), and lack of clarity in the website structure. In conclusion, she could determine that users encountered fewer usability issues in the original language version than in the localised versions (ibid.).

One year later, Pontus (2019) evaluated the degree of accessibility of a sample of 50 airline websites by performing both automated and manual checks. She noticed several general accessibility issues, mainly related to online forms, ticket purchasing, and the reservation system. In addition, she encountered more accessibility problems in the French and German (localised) versions than in the original English versions, as English "values tend to 'travel' to localised sites in other languages and create language-related accessibility problems"(ibid., 112). Overall, these two recent investigations have shown that accessibility continues to be an issue in the multilingual web and hence motivates the pertinence of yet another study in this domain.

2.3.2 Accessibility features

Torres del Rey and Morado Vázquez (2019) contributed to the research currently being conducted on localisation and accessibility by asking the following question: "does accessibility have a concrete form or clearly defining characteristics, and can the forms and characteristics that are culture- and language-bound be "captured" formally?" (ibid.). Accessibility can be seen as a 'quality' (Jiménez-Crespo 2013, 126-31; Rodríguez Vázquez 2016, 62-4; Torres del Rey & Morado Vázquez 2019), and as such, it cannot be transferred. However, as we saw in Section 2.2.1, there are a series of recommendations that can be implemented to achieve the same or even a better level of accessibility. As a result, we could state that localisers could indeed transfer these "characteristics", embedded in the form of coding elements, attributes, text units, and relations that help achieve this quality (Torres del Rey & Morado Vázquez 2019). In the context of this thesis, this is what we call 'accessibility features'.

The authors continue by making a distinction between 'neutrally transferable' (embedded in the code's structure) and 're-placed' features (in-line formatting), which are usually excluded

or protected from editing by the localisation tool. Nevertheless, localisers should take them into account, as sometimes these features need to be adapted to the target locale or simply translated (ibid.).

The following examples, taken from the home page of the Faculty of Translation and Interpreting's website[9] will help us illustrate the concept of accessibility features: we will show how they are visually displayed in the website and how they are embedded in the code.

In this first example (see Figure 2.2), we see several logos displayed towards the end of the page that represent the networks that the Faculty is part of. An individual with a visual impairment cannot see these images and, if the right technique is not implemented, will not access this information. In this specific case, by looking at the corresponding code, we notice that the web developer included the `alt` attribute, which describes the image and can be detected by a screen reader. We can consider the `alt` attribute and its value as an accessibility feature of this webpage.

Figure 2.2: Example of alternative texts.

Similarly, in the example shown in Figure 2.3, we can see the button 'Search' in the form of a lens. By clicking on this symbol, the user will be redirected to a new page. The attribute `aria-label` describes the role of the button, which can be as well detected by a screen reader. This would be another example of an accessibility feature included in a webpage's code.

---

[9] https://www.unige.ch/fti/fr/ [12.11.2019]

Figure 2.3: Example of an ARIA-label.

2.3.3 WCAG success criteria for web localisation

As we previously mentioned, WCAG includes several requirements to determine the sufficient level of accessibility, referred to as success criteria (SC). In this section, we explore which SC are more relevant to the localisation process, as we will refer to them later (see Chapter 3) when describing the methodology of our study.

Gutiérrez y Restrepo and Martínez Normand (2010) examined the WCAG and determined which requirements were the most relevant to web content localisation. The authors focused on the first three principles (perceivable, operable, and understandable), as they believed that the fourth principle, 'robust', did not include any relevant SC (ibid.).

Concerning the first principle, 'perceivable', the key issues are related to providing alternatives for non-text elements, such as images, sound, video, and interactive controls, to make accessible the information contained in these elements to people with disabilities. The SC related to this first aspect are *1.1.1 Non-text content, 1.2.1 Audio-only and Video-only, 1.2.8 Media Alternatives*, *and 1.2.9 Audio-only*. Moreover, there are a few special cases that we shall take into account. In the case of time-based media, we shall consider including captions for people with auditory impairment (*1.2.2 Captions*) and audio description for people with visual impairment (*1.2.3 Audio Description or Media Alternative*, *1.2.7 Extended Audio Description*). The second case concerns the use of sign language and the corresponding SC *1.2.6 Sign Language*. The sequence of the content, as we will see later, can affect the meaning, so it is important to localise correctly the reading sequence (*1.3.2 Meaningful Sequence*) (ibid.).

In the same way, for the principle 'operable', the key issues are related to bypassing blocks of content, page titles, focus order, link purpose and headings. It is important to provide links to skip to specific parts of the page so that even keyboard users can navigate freely (*2.4.1 Bypass Blocks*). The criterion *2.4.2 Page Titled* refers to the title of the page, which ought to be

localised, as it carries important information that appears not only in the page itself but also as the website name and when storing bookmarks. Similarly, it is fundamental for the localiser to respect the focus order, as suggested by the SC *2.4.3 Focus Order*, as a change in the order could produce a change in the meaning and operability. Furthermore, the SC *2.4.4 Link Purpose (In Context) and 2.4.9 Link Purpose (Link Only)* describe the idea that the link ought to include information about what will happen once the user activates it. Last, headings, labels and sections can contain localisable information, as suggested in the SC *2.4.6 Headings and Labels and 2.4.10 Section Heading*, that practitioners shall consider (ibid.).

The key issues related to the third and last principle, 'understandable', are language identification, unusual words, reading level, pronunciation, error management, and help. Firstly, it is fundamental for the software to detect correctly the human language, as, for instance, the screen reader used by blind people will read the text according to the language included in the tag. The SC related to this aspect *are 3.1.1 Language of Page* and *3.1.2 Language of Parts*. In the text, there could be unusual words and abbreviations that the user doesn't know, so it is important that the expanded meaning or form included in the code, following the recommendations in the SC *3.1.3 Unusual Words* and *3.1.4 Abbreviations*, is correctly localised. Consequently, the reading level (*3.1.5 Reading Level*) ought to be respected in the target product. In some cases, there is interactive content, such as forms, that include instructions or information to understand the meaning, as suggested in the SC *3.3.2 Labels or Instructions* and *3.3.5 Help*, and it is important to transfer the useful information in the target file. Finally, the last aspect concerns errors and how the designer can input data to avoid them, as suggested in the SC *3.3.1 Error Identification, 3.3.3 Error Suggestion, 3.3.4 Error Prevention* (ibid.).

2.4 Computer-Aided Translation tools

In this last section, we analyse another key object of study in our research: Computer-Aided Translation (CAT) tools. Computer-aided (or –assisted) translation is also referred to as machine-assisted human translation (MAHT), as opposed to human-assisted machine translation (HAMT), or simply machine translation (MT). In the first scenario, translators are responsible for the translation, but they benefit from the help of computerised tools, while in the second instance, the computer translates the whole text, while the human translator can edit the final output (Bowker 2002, 4).

CAT tools became popular within the translation and localisation industry at the beginning of the 1990s, as companies had to deal with large volume projects and needed tools that could speed up the translation process (Esselink 2000, 359). CAT technology consists in "any type of computerized tool that translators use to help them do their job" (Bowker 2002, 6), and does not include only a single type of software, but a variety of computerised tools such as data-capture tools, corpus-analysis tools, terminology-management systems, translation memories, localisation and webpage translation tools, and diagnostic tools (ibid., 3-8).

TS scholars and the industry employed different terms to designate the ensemble of tools: translator's workstation (Somers 2003), translator's workbench (Quah 2006, 93-4), translation environment tools (TEnTs), and CAT Systems (Garcia 2014). Moreover, due to the central role of the translation, the term 'translation memory systems' or TM tools are often employed (Esselink 2003, 80; Hutchins 1998, 11). For the purpose of this thesis, we will use the term 'CAT tool' to define the following suite of tools, or features, that are now present in more advanced systems (Garcia 2014, 70).

> *Translation memory (TM)*. This feature, as we introduced in the previous paragraph, has a central role in the whole translation system. This type of technology, originated in the 1970s, stores segmented and aligned multilingual (source and its translation) texts in databases, which can be reused in equivalent pairs of source and target segments (Bowker 2002, 92; Garcia 2014, 71; Hutchins 1998, 11; Quah 2006, 94). The program compares new untranslated segments to the ones stored in the TM, and depending on the level of correspondence retrieved, it can offer exact or perfect matches, fuzzy matches, or no match (Garcia 2014, 72).

> *Termbase*. This feature works similarly to the TM, but at the term level. The database mainly consists of a bilingual or multilingual glossary, where the system can retrieve the equivalent term (ibid., 73).

> *Segmentation*. This process produces text units by breaking the text up, and it is needed for the TM to recover possible matches (Quah 2006, 100).

> *Alignment*. Through this process, the program aligns, namely binds, the source segment to the corresponding target segment, to create new TMs or add new entries (ibid.).

> *Quality assurance (QA)*. This feature allows the translator to avert errors, such as numbers, measurements and currency that were not correctly rendered in the target

product (Garcia 2014, 76). As we will see in future chapters, this process could also be useful in the context of accessibility.

Moreover, CAT tools can support various types of files. Therefore, the system uses *filters* to convert the source text from one format to another and extract translatable segments, which will be then presented in the editor. This feature is fundamental to our research, as it provides the possibility to work with HTML files: the webpage is "usually stripped of the HTML code leaving only the text without any graphics or formatting information" (Quah 2006, 98). In Chapter 3, we will analyse in more detail how this feature works in the two selected CAT tools.

2.4.1 Prior work on CAT tools and web localisation

CAT tools have been analysed and evaluated in numerous studies, according to several aspects. As we introduced in the previous section, TS scholars studied and illustrated the application of CAT (Bowker 2002; Esselink 2000; Melby 2006; Somers 2003), while other authors highlighted the use of these tools for pedagogy purposes (Alcina 2008; Kenny 1999; Yao 2017). Moreover, other studies focused on testing and evaluating CAT tools to determine their productivity (Federico, Cattelan & Trombetti 2012), usability (Krüger 2016), or the influence on the final translation to investigate possible advantages and shortcomings (Doherty 2016; Torres Hostench et al. 2010). However, for the purpose of this thesis, we will focus primarily a number of studies that examined CAT tools in the context of web localisation, or in relation to accessibility.

First of all, we deemed it relevant to mention Morado Vázquez's work on the transfer of metadata. In her doctoral thesis, Morado Vázquez (2012) investigated the "influence that translation suggestions' provenance metadata has in the behaviour of human translators during their work when using Computer-Assisted Translation Tools" (ibid., 8). Her research focused primarily on the main localisation data exchange standard: XLIFF (XML Localisation Interchange File Format). This specific type of file format stores "localizable data and carry it from one step of the localization process to the other, while allowing interoperability between and among tools" (OASIS 2014). Through her research, she demonstrated that metadata does not have any impact on the translators' behaviour during the translation process, but the use of CAT tools (specifically of TMs) can have a positive impact on their work and on the final quality of the product itself (Morado Vázquez 2012, 261-3). In our research we will focus on a different aspect, namely specific features of HTML5; however, both researches want to

contribute in demonstrating the usefulness of the tool in showing the information (in our case, regarding accessibility) and improve the localiser's work.

As Sandrini (2008, 16) stated, web localisation needs a tool that can separate the HTML code from the text, has a translation memory, and can assist users to edit Web documents, namely HTML or XML. He continued by adding that "[a] good tool will recognise and protect all the tags in the Web page. It should, however, highlight all the elements which should be translated" (ibid., 17). As we have already seen, not all the accessibility features are included in text strings but are rather embedded in the code, and sometimes they are perceived by the tool as non-editable strings. Thus, it is crucial for the CAT tool to support these features and provide the possibility for the localiser to adapt them when needed.

As indicated by Mata Pastor (2005 in Rodríguez Vázquez 2016, 130), CAT tools do not retrieve all the translatable strings automatically, and localisers should be aware of the purpose of accessibility features and be able to, if possible, customise the tool's settings. Hence, "localisation professionals should have the necessary HTML skills to first assess which elements and attributes require their intervention, as well as an advanced knowledge of the CAT tools they are using" (ibid.).

In her doctoral thesis, Rodríguez Vázquez (2016) mentioned how CAT tools are, in fact, the most convenient tool to process files containing a markup language, such as HTML, as they can isolate translatable and localisable content from the document non-editable strings, which are usually protected to prevent any damage of the code. However, she underlined the fact that CAT tools may hide certain attributes (like the `alt` attribute) and, consequently, not show them in the editor (ibid., 126-30). As a result, if a CAT tool is not customised accordingly, certain accessibility features may not be transferred correctly in the target file.

In her master's thesis, Castro Hernandez (2015) evaluated two CAT tools: SDL Trados 2015 and MemoQ 2013. The purpose of her research was to test if CAT tools were prepared to localise files written with HTML5 new semantics (recently released at the time of the publication of her thesis). She carried out an evaluation based on the tool's effectiveness, user's satisfaction and context adequacy, characteristics included in the ISO/IEC 25010 standard. She finally determined that both CAT tools can localise HTML5-format files, and consequently, can be used for website localisation. However, she stated that some improvements could have been introduced (ibid.).

For instance, she noticed that SDL Trados Studio 2015 "did not show all the translatable text of the websites participants were required to localise" (ibid., 84). Moreover, she also stated that the same tool, compared to MemoQ 2013, provides information about the website's structural elements, which were the object of her study. On the other hand, MemoQ 2013 showed all the translatable text related to structural elements, even though, as we just said, did not add any additional information on the structure of the document. This research provides useful information for our study and lays the foundations for our work, as we will analyse the same CAT tools. Our research follows up on Castro's work by examining possible improvements in the tools' latest versions (SDL Trados Studio 2017 and MemoQ v8.7), and by analysing a concrete aspect in more detail: the processing of accessibility features.

2.4.1.1 Methods to evaluate CAT tools

Up to the present, endless CAT tool evaluations have been conducted, and many attempts to provide a comprehensive framework for evaluation have been made. In this last section, we provide an overview of translation technologies evaluation methods that were employed over the last decades.

Initially, system evaluations were requested by specific clients to determine whether the system they were interested in met with their necessities. An example of this type of evaluation is the ALPAC report, published in 1966, which focused on the status of machine translation. Later, the focus shifted on more economic aspects: the purpose of the evaluation was, in this case, to determine the commercial value of the tool. With this change came also the concept of 'context': a single evaluation was designed and carried out for a particular system, as it was going to be used in a specific context (King 2005, 45-6). Yet, these types of evaluation were not "easily reproducible since they do not supply a set of parameters which could serve as a comprehensible model" (Rico 2001).

Hence, as the variety of tools and possible scenarios kept increasing, the need for the establishment of a standard evaluation method that could be reproduced emerged. We can identify two main approaches that have been followed in recent years: a diachronic approach, based on quality management; and an approach based on quality models, in which quality is seen as a quantifiable concept (Flórenz & Alcina 2015, 82). In this thesis, we will focus on the latter, which proposes pre-normative guidelines on how to carry out the evaluation.

Among numerous project and initiatives, we find the EAGLES (Expert Advisory Group on Language Engineering Standards) evaluation group, founded by the European Union in 1993

and based on ISO work on standardisation (EAGLES 2019). In Chapter 3, we will illustrate in greater detail this approach, and how we employed it for our evaluation design.

The applicability of this approach was demonstrated by Starlander and Morado Vázquez (2013) in a paper presenting a case study carried out during the Computer-Assisted Translation Master course at the Faculty of Translation and Interpreting of the University of Geneva. In this case study, students were asked to evaluate a CAT tool by following the seven steps proposed by EAGLES. The authors highlighted how the majority of students thought that this specific evaluation method could be useful to enhance their professional background (ibid.). In a follow-up article, Starlander (2015) presented another simplification of the EAGLES approach based on the quality model introduced in the new ISO 25000 series (ISO/IEC 2014), focusing on the quality in use characteristics, which includes two additional sub-characteristics (effectiveness and satisfaction) (Starlander 2015). In Section 3.4.1, we will illustrate in details this new standard series and how it applies to our research.

The EAGLES approach was included in numerous studies that address the issue of translation technologies evaluation. Rico (2001) proposed a final user-oriented model, intending to define a comprehensive model that could be re-used in various scenarios. She established her methodology on the aforementioned approach and the ISO 9126 (ISO 2001) standard's six quality characteristics: functionality, usability, maintainability, reliability, efficiency, and portability (Rico 2001).

Moreover, in her thesis, Gow (2003) established the framework of the evaluation on this model and adapted it to the context of her research, by analysing a specific feature included in TM tools (automatic search-and-retrieval function) (ibid.). Similarly, as we explained in the previous section, Castro Hernandez (2015) decided to base her CAT tool evaluation on three characteristics included in the quality in use model of the ISO/IEC 25010 (ISO/IEC 2011) standard (effectiveness, satisfaction, and context coverage) (Castro Hernandez 2015). These investigations inspired our experiment, which will be explained in length in the following chapter.

3 Methodology

This third chapter provides a detailed description of the methodology introduced in the first chapter. In Section 3.1, we present the research questions and hypotheses illustrated in Section 1.3 as the foundation of our research, and we describe how they guided us in designing our study. In Section 3.2, we present the materials that we used for both the tool descriptive analysis and the user evaluation, focusing on the selection of the relevant SC that led to the selection of the test webpage. The following two sections (3.3 and 3.4) deal with the two main approaches adopted in our research: the first presents the descriptive approach, namely the tools descriptive analysis, in which we introduce the two selected CAT tools that we will evaluate; the second provides more information about the tool evaluation, EAGLES and the ISO standards, the participants, and the evaluation design.

3.1 Research questions and hypotheses

In Section 1.3 of the introductory chapter, we briefly illustrated the main research questions and hypotheses. Numerous factors can influence the choice of the method to adopt, such as the main objective of the study, time constraints, funding, the tool used by the participants, and the researcher's experience (Lazar, Feng & Hochheiser 2017, 25). As previously mentioned, the main purpose of our research is to analyse the impact that CAT tools have on the degree of accessibility achieved during the localisation of an HTML-format file, in which web content is usually stored.

The first research question is as follows:

> **RQ1.** Can CAT tools support and therefore transfer all the relevant accessibility features when processing an HTML5 file?

Through this first question, the researchers aim to determine whether CAT tools present features and functionalities that could contribute to the localisation of the accessibility information. As illustrated in the related work on multilingual web accessibility (see Section 2.3.1), there are several issues encountered in localised multilingual websites, specifically concerning accessibility. Because of this aspect, combined with the fact highlighted by Castro Hernandez (2015) that CAT tools (such as SDL Trados Studio) may not show some of the localisable content, we formulated the following hypothesis:

> **H1.** CAT tools do not support and transfer all the accessibility features.

The second research question is directly linked to the first one, and was formulated as follows:

RQ2. Is the resulting target file accessible?

The purpose of this second question is to examine the **functional suitability** of the tool; more specifically, we aim at studying the degree to which the two CAT tools offer correct results in localising accessibility features (*functional correctness*). As we will illustrate in greater detail in Section 3.4.1, *functional correctness* is one of the three sub-characteristics of *functional suitability*, one of the eight characteristics that constitute the software product quality included in the ISO/IEC 25010 standard (ISO 2011). Together with *functional completeness* and *functional appropriateness*, *functional correctness* represents the dependent variables of our study. Dependent variables, as stated in Lazar et al. (2017), are the main interest of the researcher, the outcome or the effect that the researcher wants to investigate. Likewise, we can define independent variables as the factors that can have an impact on the dependent variable, namely the variable that the researcher manipulates (ibid., 30). In our specific case, we will manipulate two independent variables. The first is the CAT tool that we will use to process the HTML file.

To design the experiment, we formulated the following hypothesis:

H2. The functional completeness of the tool used has an impact on the final degree of the accessibility achieved.

H3. The functional appropriateness of the tool used has an impact on the final degree of accessibility achieved.

In the previous chapter, we also stated that localisers' degree of knowledge of accessibility best practices can influence the final achievement of accessibility in the target product. To analyse the effect of this second independent variable (i.e. knowledge of accessibility), we formulated the fourth and last hypothesis:

H4. The participants' level of knowledge of accessibility has an impact on the final degree of accessibility achieved.

3.2 Material selection

For both the descriptive approach and the tool evaluation, we decided to process an accessible HTML5 code. We chose and adapted the final code according to the following steps: (1) selection of a set of requirements, concretely a set of SC, that help us determine the features that we want to take into consideration and that we will later examine in the target product; (2)

selection of an accessible website, based on several criteria; (3) selection of two relevant webpages comprising all the selected requirements; and (4) adaptation of the final code, according to the researcher's needs.

3.2.1 Selection of WCAG success criteria

To determine the metrics and the method to be applied in the system evaluation, it is necessary to establish a set of requirements. Within the framework of our thesis, we considered that the best way to define these requirements was by selecting a limited number of SC from the WCAG that are relevant to both localisation and accessibility. In Chapter 2, we already illustrated the SC that Gutiérrez y Restrepo and Martínez Normand (2010) deemed pertinent to both fields, which are included in the first three principles of the WCAG (perceivable, operable, and understandable). Similarly, Torres del Rey and Morado Vázquez (2019) considered the same three principles for the selection of eleven SC that are directly related to localisation. The purpose of their study concerns the possibility of conveying accessibility through localisation and internationalisation standards (ibid., 1). They focused on two specific standards, which can be used in the preparation of original materials and the document transformation during the localisation process: Internationalisation Tag Set (ITS), and XLIFF, that we introduced in Section 2.4.1. The latter is also relevant to our research, as both SDL Trados Studio and MemoQ convert the source-format file (in our case, HTML) into an XLIFF document during the localisation process. Interoperability (which define the ability of the XLIFF file to work with different systems) is a key aspect of this type of file format, as it allows the extracted content of HTML documents to be represented identically in the XLIFF document, although different tools may provide different filters to extract the content (OASIS 2006).

As stated in Section 2.4.1, we are not going to focus on XLIFF documents but on a number of specific HTML5 features. However, we deem it relevant to illustrate briefly the localisation process that includes XLIFF as an intermediary format (Figure 3.1). The first step includes the separation, or extraction, of the translatable text from the document's skeleton (layout data), which will be subsequently stored in a bilingual XLIFF document. The original text strings, extracted by the tool, will be stored under the `source` element, while the translation will be retained under the `target` element. The second step consists of the actual localisation and translation process. Once the translation is finalised, the tool will reconvert the bilingual XLIFF file into the source format (merging) and will export it (Bordandini 2019, 14; Raya 2004).
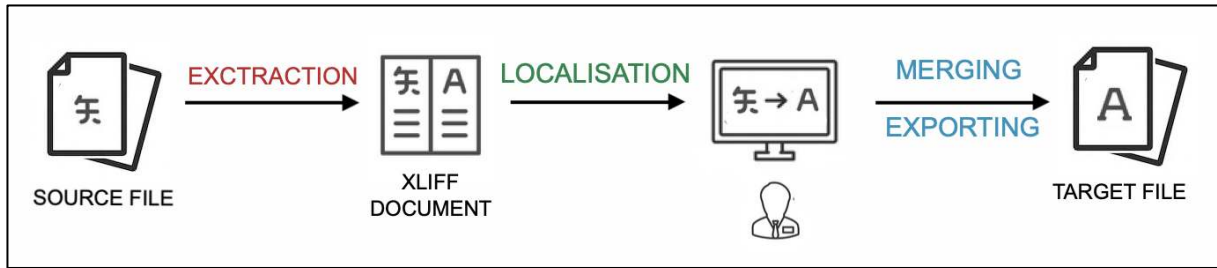
Figure 3.1: Localisation process with a CAT tool.

Torres del Rey and Morado Vázquez (2019) also highlighted an important aspect that it is pertinent to our research: the content and information included in the source code can be transferred in an XLIFF document, however, the tool used to extract the information "should recognize the localizable information […] and include it in the XLIFF document that will later be manipulated in a CAT tool by a localizer" (ibid., 14). Therefore, the same principle can be applied to HTML files: if the tool does not correctly extract the localisable content, the practitioners may not be able to localise the content.

As both Torres del Rey and Morado Vázquez (2019)'s study and our research deal (although, in our case, only in part) with this specific type of document, and their study also focuses on the transfer of accessibility features, we selected a subset of the eleven SC chosen by said authors. They considered SC that (1) relate to language, text or localisable modes or cultural assets, (2) refer to basic functionality and affordances, and (3) are fundamental to understand the content of the page. The SC covered in their preliminary study are: *1.1.1 Non-text Content*; *2.4.1 Bypass Blocks*; *2.4.2 Page Titled*; *2.4.3 Focus Order*; *2.4.4 Link Purpose (In Context)*; *3.1.1 Language of Page*; *3.1.2 Language of Parts*; *3.2.1 On Focus*; *3.2.2 On Input*; *3.3.1 Error Identification*; *3.3.2 Labels or Instructions* (ibid., 12-3).

Ultimately, although structural elements can offer additional information on the functional and situational context of the document, we opted to focus exclusively on the localisable content in the HTML file rather than the information that is neutrally transferred. For this reason, we chose not to consider the two SC *2.4.3 Focus Order* and *3.2.1 On Focus*. The final SC that constitute the requirements for our study are the following nine SC (Table 3.1):

Success Criterion	Guideline	Principle
1.1.1 Non-text Content	1.1 Text Alternative	1. Perceivable
2.4.1 Bypass Blocks	2.4 Navigable	2. Operable
2.4.2 Page Titled	2.4 Navigable	2. Operable
2.4.4 Link Purpose (in Context)	2.4 Navigable	2. Operable
3.1.1 Language of Page	3.1 Readable	3. Understandable
3.1.2 Language of Parts	3.1 Readable	3. Understandable
3.2.2 On Input	3.2 Predictable	3. Understandable
3.3.1 Error Identification	3.3 Input Assistance	3. Understandable
3.3.2 Labels or Instructions	3.3 Input Assistance	3. Understandable

Table 3.1: Selected SC.

3.2.2 Selection of the test webpage

Once we determined the SC to be studied, the next step concerned the selection of the test webpage. As accessibility requirements and related regulations have been introduced in several countries' legislations[10], which often imply that all the official websites in the public sector shall be barrier-free, we decided to choose a few governments' websites as a starting point and proceed with a manual inspection to determine which one satisfied all the requirements mentioned above. Eventually, two webpages on the Government of Canada's official website, specifically on the Youth Council section[11], were selected. The first one consists of the 'April 2018 Youth Newsletter' (Figure 3.2)[12], while the second is directly linked to the first page and comprises a form to sign up and engage with the Youth Council[13]. For the specific purpose of our study, the latter was added to the test webpage because it includes two SC that are missing on the first page: *3.2.2 On Input*, and *3.3.1 Error Identification*.

---

[10] For instance, in the United States, web accessibility is regulated in the Rehabilitation Act in 1998. Switzerland included a few regulations concerning the elimination of inequalities in the art. 8 of the Federal Constitution, while the European Union has ratified the United Nations Convention on the Rights of Persons with Disabilities (CRPD) adopted in 2006.

[11] https://www.canada.ca/en/youth.html [19.11.2019].

[12] https://www.canada.ca/en/youth/services/youth-newsletter/2018-april.html [19.11.2019].

[13] https://www1.canada.ca/en/contact/youth_engagement.html?utm_source=ynlen [19.11.2019].

Figure 3.2: Page of the Youth Newsletter April 2018.

Two main reasons are behind the researcher's decision for selecting the Government of Canada's website. The first is based on legal requirements. Until this year, web accessibility was not regulated by any federal law. Nevertheless, through the years, several provinces enacted accessibility legislation: in Ontario, the *Accessibility for Ontarians with Disabilities Act (AODA)* became law in 2005; the Government of Manitoba enacted the *Accessibility for Manitobans Act (AMA)* in 2013; and in 2017, Nova Scotia's *Accessibility Act* was introduced. In 2018, *The Accessible Canada Act: An Act to Ensure a Barrier-free Canada* (ACA) was proposed by the Minister of Science and Sport and Persons with Disabilities and received Royal Assent (and consequently became law) on the 21st June 2019 (Doyle 2019). One of the main objectives of the bill is to ensure that barriers in information and communication technologies are prevented (ACA 2019). Moreover, Canada had already introduced the 'standard on web accessibility', which took effect in 2011 and ensures that "a high level of Web accessibility is applied uniformly across Government of Canada websites" (Government of Canada 2011). This specific standard ensures that all webpages meet the five WCAG 2.0 conformance requirements: conformance level (AA), full pages, complete processes, only accessibility-supported ways of using technologies, and non-interference (ibid.).

The second reason concerns the fact that Canada, as a bilingual country, must ensure that all webpages the institution is accountable for are available in the two official languages (English and French) (Government of Canada 2012, art. 6.6). This will provide useful support to the researcher, as the French version of the two webpages' code can be used as a reference document and can be compared to the target files exported from the two CAT tools when examining the evaluation results.

The last step of the test webpage creation process concerns the adaptation of the final code. As mentioned above, the second page was combined with the first one because of two missing SC on the first page. Since some parts, mostly in the header section (including menu, bypass blocks

elements, language selection, etc.) and in the footer section (error report form, copyrights, social network links, etc.) are basically the same across both web pages, we added only the code of the form (with the title and the description to avoid deleting useful information about the context) to the code of the first page. We inserted it before the footer section to maintain a logical structure order. Moreover, as the newsletter included several sections, we decided to remove the most repetitive articles. The final structure of the test webpage is as follows: title and page description; header section; articles (4); sign up form; copyrights; error report form; footer section (see Appendix B). In the following section, we will illustrate several examples of accessibility features in the selected test webpage and describe the techniques used.

3.2.3 Accessibility features in the selected test webpage

As mentioned in Section 2.2.1, in the WCAG we find different types of techniques that can be applied to achieve accessibility. The techniques we considered are General, ARIA, and HTML techniques. On the contrary, we did not take into consideration CSS techniques as we will not be concerned by the webpage appearance (and, in our case, the CSS information is stored in another file). We also did not include Flash, PDF, Silverlight, and SMIL techniques as they deal with different types of documents or technologies. Eventually, we also excluded Client-Side Script and Server-Side Script as they are not directly linked to localisation. Moreover, we decided not to include advisory techniques and failures but focus on sufficient techniques only.

3.2.3.1 Non-text Content

The intent of the SC *1.1.1 Non-text Content* is to make the information included in non-text content (such as images, time-based media, controls or input) accessible through the use of text alternative (Kirkpatrick et al. 2018). In the following two examples (Figure 3.3 and Figure 3.4), web editors employed the technique *H37: Using alt attributes on img elements*. In the first case, the alt attribute describes what it can be seen in the picture, while in the second example it also adds information about the content of the video linked to the image.

Figure 3.3: Example of text alternative used to describe an image.

Figure 3.4: Example of text alternative used to describe the media content.

3.2.3.2 Bypass Blocks

Through this mechanism, the user can access directly to the primary information by skipping the repetitive content included in other webpages of the same website. In our test webpage, the technique used is the *G1: Adding a link at the top of each page that goes directly to the main content area*, which allows the user to skip repetitive blocks. This technique mainly benefits people with visual impairments and people with cognitive limitations (W3C/WAI 2019d). As we can see in Figure 3.5, these links are not visible on the page but are embedded in the code.

Figure 3.5: Example of a link to bypass blocks of repetitive content.

3.2.3.3 Page Titled

The purpose of the title is to offer information about the main content of a webpage (Kirkpatrick et al. 2018). Without this factor, users can not orient themselves while navigating the webpage. The techniques implemented in the test webpage are *G88: Providing descriptive titles for Web pages* and *H25: Providing a title using the title element*, as seen in Figure 3.6. The `title` attribute will also be displayed in the title bar.

Figure 3.6: Example of a `title` attribute.

3.2.3.4 Link Purpose (in Context)

Links are the fundamental concept of the Internet; they connect webpages and allow the user to navigate through different websites. However, if the purpose of the link is not clear, the user will not be able to understand where it directs to. The purpose can be determined by the link text and its pragmatically determined link context (ibid.). In our case, the techniques used are *G91: Providing link text that describes the purpose of a link* and *H30: Providing link text that describes the purpose of a link for anchor elements* (Figure 3.7).
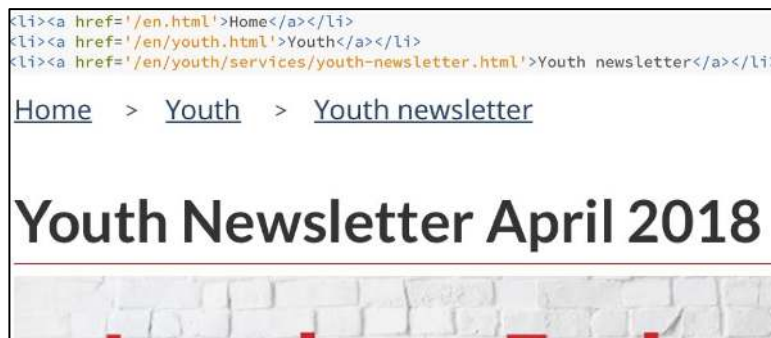
Figure 3.7: Example of three `a` anchor elements.

3.2.3.5 Language of Page

The purpose of this SC "is to ensure that content developers provide information on the Web page that user agents need to present text and other linguistic content correctly. Both assistive technologies and conventional user agents can render text more accurately when the language of the Web page is identified. Screen readers can load the correct pronunciation rules. Visual browsers can display characters and scripts correctly. Media players can show captions correctly. As a result, users with disabilities will be better able to understand the content." (W3C/WAI 2019e). Therefore, this SC is fundamental for the proper functioning of user agents. In the context of our test webpage, the technique *H57: Using the language attribute on the*

37

HTML element was implemented. As we can see in Figure 3.8, the `lang` attribute includes the language of the page (English).

```
<!--[if gt IE 8]><!-->
<html class="no-js" dir="ltr" lang="en" xmlns="http://www.w3.org/1999/xhtml">
<!--<![endif]-->
```

Figure 3.8: Example of `lang` attribute.

3.2.3.6 Language of Parts

Similarly, this SC consists of defining the language of parts that are in another language other than the language of the page, except for proper names, technical terms, etc. (Kirkpatrick et al. 2018). In our specific case, although there are some parts in French, no technique was implemented. However, as we also want to examine this aspect, we decided to apply the technique *H58: Using language attributes to identify changes in the human language* to see if CAT tools can support this feature. Figure 3.9 shows the language selection option in the header section, while Figure 3.10 consists of the French title of the same webpage. In both cases, the research added the `lang` attribute.

```
<a lang="fr" href="/fr/jeunesse/services/infolettre-
jeunesse/2018-avril.html">
    Fran&ccedil;ais
```

Français

.ca   🔍

Figure 3.9: Language selection option.

```
<p>Irregular.<br>
    Title in French: <title lang=fr>Infolettre jeunesse</title>.<br
Available also on the Internet: <a href="https://www.canada.ca/yout
target="_blank">www.canada.ca/youth</a> <br>
```

Irregular.

Title in French: Infolettre jeunesse.

Available also on the Internet: www.canada.ca/youth

Figure 3.10: Webpage title in French.

3.2.3.7 On Input

The main intent of this SC consists of making sure that the user is aware of the effects when entering data or selecting a form control. Changes in context can be a source of confusion for people with disabilities that are not able to perceive the change, therefore changes should only be the consequence of the user's action (W3C/WAI 2019f). In the test webpage, this feature is included in the sign-up form and it has been implemented through two techniques: *G80:*

Providing a submit button to initiate a change of context, and *H32: Providing submit buttons* (Figure 3.11).

Figure 3.11: Example of submit button.

3.2.3.8 Error Identification

When an error occurs, the user should be able to determine what is wrong. Hence, the error should be first identified and then described in the form of text (Kirkpatrick et al. 2018). In order to show the error in our test webpage, we proceeded to submit the form without filling in any field. As a result, several errors were identified and, only then, we saved the source code. As we can see in Figure 3.12, the technique *G83: Providing text descriptions to identify required fields that were not completed* was implemented.

Figure 3.12: Example of error identification.

3.2.3.9 Labels or Instructions

Through this SC, web editors can provide information about the webpage's content: "[t]he intent of this Success Criterion is not to clutter the page with unnecessary information but to provide important cues and instructions that will benefit people with disabilities" (W3C/WAI, 2019g). For this specific feature, we will mainly focus on ARIA labels. Through this attribute, the content's author can offer additional instructions that will not be visible on the webpage but will be detected by assistive technologies. Figure 3.13 illustrates the techniques implemented in this specific case: *G131: Providing descriptive labels* and *ARIA9: Using aria-labelledby to concatenate a label from several text nodes*.

Figure 3.13: Example of `aria-label` attribute.

3.3 Tool descriptive analysis

To answer the first research question RQ1 we decided to adopt a descriptive approach, which enables us to observe and describe how the two CAT tools deal with accessibility features. The descriptive approach is often considered as the first step of research: this method serves as a starting point to describe the actual situation of a given area of study and allows researchers to develop subsequent research stages (Rodríguez Vázquez 2016, 18-9). In other words, descriptive research approach can be defined as the basic research method that "involves identification of attributes of a particular phenomenon based on an observational basis, or the exploration of correlation between two or more phenomena" (Williams 2007, 66). Therefore, we can state that the systematic and accurate description of the research's object is the central aspect of this particular research design, as it provides researchers with tools to identify new meanings, describe what already exists, and categorising information (Dulock 1993, 154).

Concretely, in the context of our research, we will carry out an observational study of the two CAT tools during the localisation process. We will work with the test webpage that we have previously described and we will observe how the tools process the specific features embedded in this type of file. We will take into consideration several aspects: (1) the specific functionalities provided by the tool related to both the file format that we are processing (i.e. eventual tool settings that are related to HTML) and accessibility; (2) the actual support of the accessibility features included in the requirements and embedded in the code; and (3) how these features are displayed in the tool's editor. Concerning the second and third aspect, we will examine if features are effectively shown to the localiser and, if so, whether the tool presents further information on the feature's context (i.e. tags including the protected part of the code; use of different typographical emphasis, etc.).

Hence, this first step of our research will enable us to lay the foundations for the second part: the data and results collected in the tool descriptive analysis will be further considered for the analysis of the tool evaluation results. In the following sub-sections, we will introduce the two CAT tools observed in this research: SDL Trados Studio 2017 and MemoQ v8.7.

3.3.1 SDL Trados Studio 2017

SDL Trados Studio is one of the most popular CAT tools on the market. This TM software is a complete translation environment that enables translators and localisers "to edit and review projects, use agreed terminology and leverage machine translation results" (SDL Trados, 2017a), and offers several functionalities such as TM, Termbase, automatic QA function, MT, and many others. Among the most recent updates, SDL Trados Studio introduced a project manager (PM) functionality (combined with SDL Trados GroupShare) and the possibility to manage the terminology in the cloud (SDL Trados 2017a; 2017b). Although a more recent version (SDL Trados Studio 2019) has been released, for reasons of availability we will use the previous version: SDL Trados Studio 2017. However, this factor should not have a major impact on the results, as the new features included in the latest version do not concern directly the features and functionalities investigated in this research[14].

Compared to the version of 2015 tested in Castro Hernandez's (2015) thesis, numerous features, enhancements, and other changes have been introduced in Studio 2017 including, in addition to the aspects already mentioned, features to speed up the translation process (LookAhead), support for Google Neural MT, new user interface, etc. Among these, we deemed it relevant to our research to point out the implementation of file types improvements, specifically of HTML-format files: the issue concerning hyperlink handling inside hidden text was improved; while problems concerning the incorrect encoding for HTML files and Writer settings for HTML embedded content processor were fixed (SDL Trados 2017a; 2017b).

3.3.2 MemoQ v8.7

Likewise, MemoQ is a translation management system (TMS) and includes similar features and functionalities of SDL Trados Studio: TM, Termbase, LiveDocs, built-in QA function, MT, and integration with TransPDF to allow the translation of PDF documents. In our research, we will test the latest version of the CAT tool, MemoQ v8.7, which introduced several features compared to the version used in Castro Hernandez's thesis (MemoQ 2013). The updates

---

[14] The changes introduced in SDL Trados Studio 2019 include integration between Language Cloud Translation Management and SDL Trados Studio; support for secure workflows in Trados Studio, file types enhancements; etc. Moreover, the file types enhancements do not include HTML-format files.

[https://docs.sdl.com/LiveContent/web/pub.xql?action=home&pub=SDL%20Trados%20Studio%20Help-v5&lang=en-US#docid=GUID-E8846956-386C-4F4D-A81E-8E097E196366&addHistory=true&query=&scope=&tid=&filename=GUID-E8846956-386C-4F4D-A81E-8E097E196366.xml&resource=&inner_id=&toc=false&eventType=lcContent.loadDocGUID-E8846956-386C-4F4D-A81E-8E097E196366, 24.11.2019].

comprise enhanced leverage, improved compatibility with other tools, improved translation management, keyboard shortcut editor, the introduction of a project management dashboard, and web-based project and user management. Moreover, in the context of our research, we noted a recent improvement concerning the HTML filter, in which the text is segmented along tags (MemoQ 2018; 2019).

3.4 User evaluation

An evaluation of the two CAT tools functional suitability was deemed the best method to answer the second research question RQ2. Through this experiment, the researchers aim to measure three dependent variables (functional correctness, functional completeness, and functional suitability) to examine the effect produced by two independent variables (CAT tools and participants' knowledge of accessibility). In other words, we will manipulate the two CAT tools and the knowledge of accessibility (by dividing the participants into two groups) to examine the effect on the three sub-characteristics of functional suitability. Concerning the design of the evaluation, we based the experiment on the EAGLES method and the ISO/IEC 205010 standard, which we briefly introduced in Section 2.4.1.1 and we will further expand on in the following section.

3.4.1 EAGLES method and ISO standards

EAGLES was established in 1993 with the purpose of responding "to the lack of common technologies and standards for the language industries" (Höge 2002, 38). The '7-step recipe' on which we based the design of our evaluation was published in 1999 and includes the following steps: (1) define the reason behind the evaluation, understand why the evaluation is being done; (2) elaborate a task model to identify all the relevant roles and agents; (3) define the top level quality characteristics, specifically the features that we want to evaluate; (4) produce detailed requirements and define a valid method to measure the performance of the object being evaluated; (5) devise the metrics to be applied to the system for the selected requirements; (6) design the evaluation, its tasks and possible scenarios; (7) execute the evaluation and summarise the results (EAGLES 1999). An important aspect, defined as the golden rule by Rico (2000), is to determine the context of use before carrying out the evaluation (EAGLES 1999; Rico 2000, 36).

As stated in the Evaluation of Natural Language Processing Systems final report (EAGLES 1996, 11), the EAGLES Evaluation Working Group initially based its work on the ISO 9126 standard (ISO 2001), which establishes a software quality model and defines its six

characteristics (functionality, usability, maintainability, reliability, efficiency, and portability). Through a quality model, we can determine which characteristics we need to take into consideration when evaluating the software's features and functionalities, and which characteristics are pertinent to the context of use of our study. Besides the ISO 9126 standard, we need to mention another standard that was also relevant to the EAGLES method, namely the ISO 14598 standard (ISO 1999), which described activities to perform the evaluation and was later revised by the ISO/IEC 25040 standard (ISO/IEC 2011a). The latter is part of the ISO/IEC 25000 series of standards (ISO/IEC 2014), also known as SQuaRE (System and Software Quality Requirements and Evaluation), which is the evolution of the two abovementioned standards. Among the numerous standards included in this series, we will focus on the ISO/IEC 25010 (ISO/IEC 2011).

The new standard is divided into two quality models: software *quality in use*, which includes effectiveness, efficiency, satisfaction, freedom from risk, and context coverage; and software *product quality*, which consists of reliability, performance efficiency, compatibility, usability, functional suitability, security, maintainability, and portability (Figure 3.14). These characteristics "relate to static properties of software and dynamic properties of the computer system" (ISO/IEC 2011).
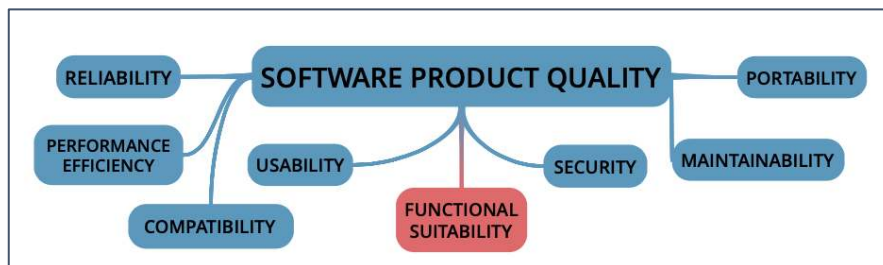
Figure 3.14: Characteristics of the ISO/IEC 25010 product quality.

As we previously explained, through the tool evaluation, we want to determine the functional suitability of the CAT tools. With the term *functional suitability*, we understand the product quality characteristic that "represents the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions" (ISO 25000 2019). In other words, through this characteristic, we are able to indicate the tool's level of fulfilment of the selected requirements, which "helps ensure that the software product is suitable for the functions it must perform" (Rodríguez, Oviedo & Piattini 2016, 19). It is composed of three sub-characteristics (illustrated in Figure 3.15): *functional correctness*, which determines the degree to which the tool enables the user to obtain valid results; *functional*

completeness, which defines the degree to which the functions of the tool cover all the required tasks and objectives; and *functional appropriateness*, namely the degree to which the tool's functions and features help the user accomplish the tasks (ISO 25000 2019).
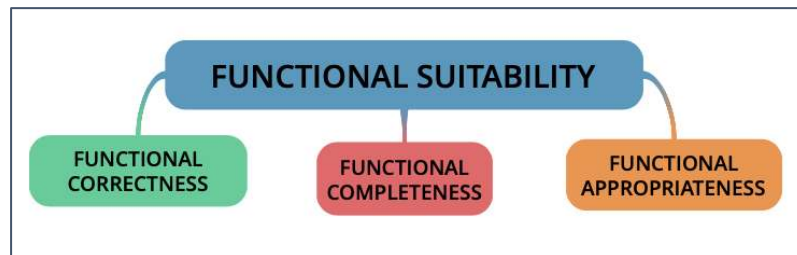
Figure 3.15: Sub characteristics of functional suitability.

Within the framework of our research, we interpret these three sub-characteristics as follows:

a) *Functional correctness*: the degree to which the system produces an accessible target file. To evaluate this sub-characteristic, we will analyse the target HTML code produced by all the participants and verify that all the necessary changes have been made.

b) *Functional completeness*: the degree to which the tool's features and functionalities cover all the tasks, namely the localisation of all the SC and subsequent techniques mentioned in Section 3.2.1 and 3.2.3.

c) *Functional appropriateness*: the degree to which CAT tools facilitate the accomplishment the abovementioned tasks. We will examine this aspect by reviewing the participants' answers to a post-evaluation questionnaire.

3.4.2 Participants

Another factor that had an influence on the evaluation design was the participants' profile. Ten participants in total took part in the experiment and, before carrying out the actual evaluation task, they were asked to complete a questionnaire (see Appendix C), which included several background questions to verify their knowledge on localisation, accessibility, CAT tools, and HTML. Once the answers to the background questionnaire were analysed, we outlined the participants' profile and eventually reached the conclusion to add a supplementary hypothesis related to the participants' level of knowledge of accessibility (H4).

To recruit participants for the experiment, we drafted a call for participation (see Appendix D), which was posted on the forum of the Master's Localisation course and sent to current and former Master students of the Italian Unit at the Faculty of Translation and Interpretation of the University of Geneva. A description of the experiment, including the evaluation's objective, information about when and how the study will be carried out, together with a link to the

background questionnaire, were included in the document. The consent form to accept to participate in the experiment was added as the first part of the questionnaire, followed by a section comprising more general questions about their native and second languages, and their current occupation. The following part included questions about the two fundamental notions of our research: localisation and accessibility. The participants were asked whether they were familiar with both concepts, whether they had ever localised a web file, and if they ever dealt with accessibility issues during the localisation process. The last part of the questionnaire served to determine their skills concerning the object of our research: CAT tools. Thus, participants were asked whether they ever used CAT tools and, if yes, how often they use them. The following questions related to SDL Trados Studio and MemoQ, to determine their level of expertise and whether they ever used them for localisation projects. The last questions concerned HTML, intending to define whether they were familiar with this markup language and, consequently, with the file format. In the following section, we will illustrate the answers to the background questionnaire and how they helped shape the design of the evaluation.

3.4.2.1 Participants' profile

Language skills and occupation

All participants (N=10) selected Italian as their native language. Concerning passive languages, all of them (N=10) selected French, while English was indicated by the majority of the participants (N=9). However, the participant who did not select English as a passive language has sufficient knowledge of the language and can easily read and understand the content. Therefore, as we will not focus on the participants' language or translation skills per se, it should not influence the final results. Other languages were also listed: German (N=5), Spanish (N=3), and Croatian (N=1). Concerning the participants' occupation, most of them (N=9) indicated that they are currently enrolled at the Faculty of Translation and Interpretation at the University of Geneva, with only one (P07) in their first year of study while the rest (N=8) is in their second year (or more) of study at Master's level. One participant (P06) is at present working in the language engineering field.

Localisation knowledge

Concerning localisation, all participants (N=10) indicated that they are familiar with this notion. Two participants (P07, P08) specified that they are currently following the Localisation course, while three (P03, P05, and P10) indicated that they followed it in the past. Another participant (P02) mentioned that they chose to work on the localisation of a website for their Master's

thesis. Moreover, four participants (P01, P03, P04, and P07) mentioned that they already localised an HTML file, with two of them (P04 and P06) stating to have used a CAT tool.

Accessibility knowledge

On the contrary, not all the participants (N=8) indicated that they are familiar with accessibility. Two of them (P01 and P09) indicated that they do not have any knowledge about accessibility. Furthermore, only half (N=5) mentioned dealing with accessibility issues. P07 responded that they had to assess accessibility in a personal website for the Localisation course, while two other participants (P03 and P06) mentioned 'image description' and alternative texts for non-text content as examples of accessibility issues they encountered. Three participants (P02, P04, and P05) indicated that they were familiar with the notion of accessibility, but they never had to deal with any issue. The data collected concerning this specific aspect prompted us to divide the participants into two groups for the experiment: one with basic knowledge of accessibility and experience with accessibility issues, and one with no experience with accessibility issues and no or basic knowledge of accessibility.

CAT tool skills

All the participants (N=10) confirmed that they had already used a CAT tool in the past. However, the frequency of use varies: a few of them (N=3) indicated that they employ them rarely, nearly half (N=4) selected the option 'sometimes', while one selected 'often', and the rest (N=2) indicated that they use CAT tool for every translation project. Moreover, all of them stated that they used both SDL Trados Studio 2017 and MemoQ v8.7, but, even in this case, the answers about the level of familiarity with the tools varied. In general, participants considered themselves more familiar with SDL Trados Studio ($\bar{x}$=3,2) than with MemoQ ($\bar{x}$=2,8). However, both tools were used for similar types of projects: the majority (N=7) mentioned using CAT tools for general and technical translations, and more than half (N=6) also added that they used the tools for localisation projects.

HTML knowledge

All participants (N=10) answered positively to the question related to HTML ('Have you ever heard of HyperText MarkUp Language (HTML)?'). Only one participant (P09), however, mentioned to not have worked with this language or file format before. This issue can be resolved by conducting a small tutorial before the experiment, in order for the participant to get familiar with the localisation process of this specific file format. The others (N=9) all affirmed to have created a website or webpage during at least one university course (Localisation and/or

Web Technologies) and consequently modified an HTML code with a web design tool or a source code editor, such as Notepadd++, Atom, Wordpress, and XML editors.

The following table (Table 3.2) is a summary of the participants' profile. Letters N and P stand for 'native language' and 'passive language', respectively. Concerning the colours, yellow indicates the languages included in participants' language combination, blue indicates the participants' occupation, while orange shows the participants' skill for both CAT tools. Moreover, green is used to show affirmative answers to questions concerning the participants' knowledge concerning localisation, accessibility, and HTML, while red indicates negative answers.

	Languages				Occupation		Localisation		Accessibility				CAT tools		HTML	
	Italian	English	French	Others	Student	Other	Knowledge	No knowledge	Knowledge	No knowledge	Experience	No experience	SDL Trados	MemoQ	Experience	No experience
P01	N		P	yellow	blue		green			red		red	orange	orange	green	
P02	N	P	P	yellow	blue		green		green			red	orange	orange	green	
P03	N	P	P	yellow	blue		green		green		green		orange	orange	green	
P04	N	P	P	yellow	blue		green		green			red	orange	orange	green	
P05	N	P	P	yellow	blue		green		green			red	orange	orange	green	
P06	N	P	P			blue	green		green		green		orange	orange	green	
P07	N	P	P	yellow	blue		green		green		green		orange	orange	green	
P08	N	P	P	yellow	blue		green		green		green		orange	orange	green	
P09	N	P	P		blue		green			red		red	orange	orange		red
P10	N	P	P		blue		green		green		green		orange	orange	green	

Table 3.2: Summary of the participants' profile.

3.4.3 Evaluation design

In this section, we will describe how we designed the tool evaluation according to the EAGLES 7-step recipe. The first step concerns the definition of the context of use and the purpose of the evaluation. With this evaluation, we want to determine if CAT tools, specifically SDL Trados Studio 2017 and MemoQ v8.7, can process an accessible source file and, consequently, produce an equally accessible target product. For this reason, we will not evaluate the system as a whole, but we will concentrate on the related features and functionalities.

To identify the requirements we want to take into account, it is fundamental to develop a task model that will support us in understanding the roles and the processes involved in the use of the system. In order to create a specific task model for our experiment, we exploited the computer-assisted text transformation system task model (Figure 3.16) proposed in the

EAGLES final report (EAGLES 1996). The final task, in this case, is the transformation of the source text in a target language.
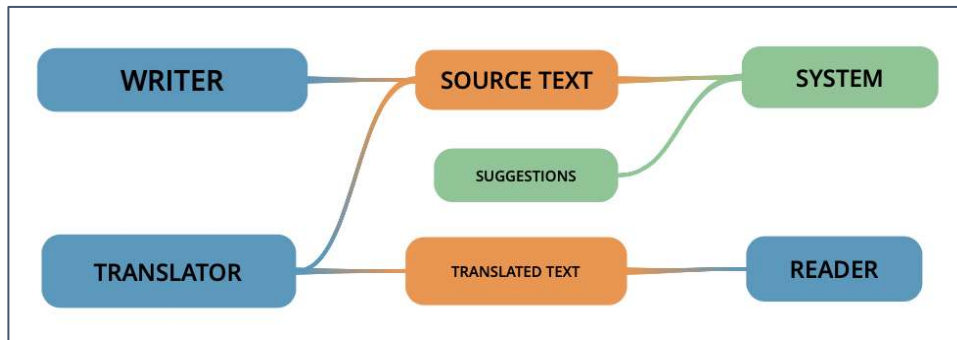
Figure 3.16: Task model (adapted from EAGLES Evaluation Working Group Final Report 1996, 28).

As both CAT tools taken into consideration for our evaluation are considered TM systems, we adapted this task model to the context of our experiment (see Figure 3.17).
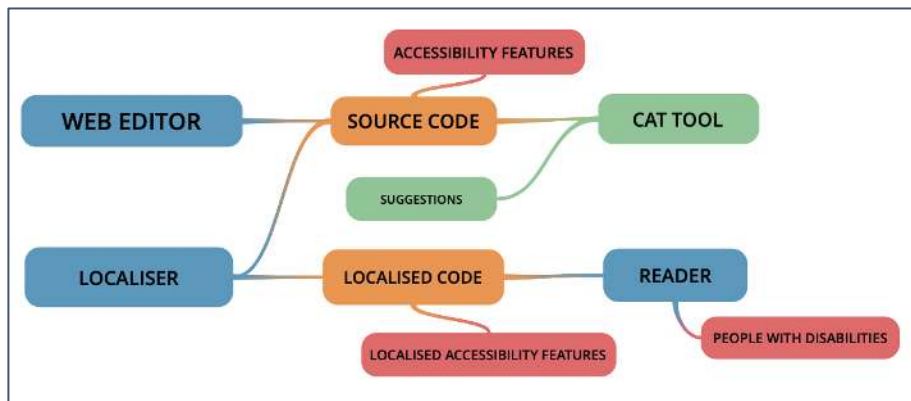
Figure 3.17: Final task model.

To better understand the role of localisers in this specific context, we use the target public (reader), which in our case includes people with disabilities, as a starting point. As people with impairments are also considered when developing the source code, the web editor decides to introduce certain accessibility features that enable handicapped people to access content online. Hence, the localiser should acknowledge the presence of people with impairments in the new target public. For this reason, the final task is to successfully transfer this specific type of information and produce an accessible localised code. When evaluating the system, localisers should question if the CAT tool provides the right functionalities and features to support and transfer accessibility features.

The third step, illustrated in Section 3.4.1, comprises the definition of top level characteristics. As we previously mentioned, we will evaluate the **functional suitability** of the tools. More specifically, we will determine the tools' functional correctness, that is, whether the accessibility features are correctly localised in the target file. We will consider functional

completeness and functional appropriateness as supplementary variables that can have an influence on the final degree of achievement of accessibility. The following step concerns the relevant requirements to determine the metrics and the method to be applied to the system, namely the SC that we selected during the material selection. The SC will serve as the measurement to determine the functional correctness of the tool, as they can be defined as 'measurable attributes'. As a result, the researcher will analyse the target code and calculate the percentage of accessibility achieved by every participant. In Table 3.3, we enumerated the SC's occurrences in the source code.

Success Criteria	N° of occurrences
1.1.1 Non-text Content	18
2.4.1 Bypass Blocks	3
2.4.2 Page Titled	1
2.4.4 Link Purpose (in Context)	46
3.1.1 Language of Page	1
3.1.2 Language of Parts	2
3.2.2 On Input	3
3.3.1 Error Identification	5
3.3.2 Labels or Instructions	1
TOTAL	80

Table 3.3: Measurable attributes.

As some SC occur more often in the document compared to other requirements, we will also consider all SC as 'tasks' that participants have to accomplish. As a result, we will be able to measure statistically the success rate for every participant and determine the mean result to calculate the degree of functional completeness of both tools (see Table 3.4).

SUCCESS CRITERIA	Non-text Content	Bypass Blocks	Page Titled	Link Purpose	Language of Page	Language of Parts	On Input	Error Identification	Labels or Instructions
TASK	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9

Table 3.4: SC divided into tasks.

Moreover, functional completeness will be measured through the integration of the results of the descriptive analysis carried out in the first part and the participants' answers to the post-evaluation questionnaire. Those answers will also be used as the tool to determine the functional appropriateness.

Concerning the actual evaluation task, the participants were asked to process the test webpage with both SDL Trados Studios 2017 and MemoQ v8.7. They were divided into two groups: one with basic knowledge of accessibility and experience with accessibility issues (Group 1), and one with no experience with accessibility issues and no or basic knowledge of accessibility (Group 2).

Group 1	Group 2
P03	P01
P06	P02
P07	P04
P08	P05
P10	P09

Table 3.5: Groups for the experiment.

The two groups were asked to carry out the same exercise (i.e. to localise the test webpage): once they added the source file, they were instructed to eventually modify the settings they believed concerned accessibility and, only then, localise the file (see Appendix E). We have, however, two different scenarios. The first involves novice web localisers with knowledge on web accessibility, who are sufficiently aware of the possible issues that they can encounter and the hypothetical problems they ought to resolve. By giving Group 1 the possibility of making adjustments to the tool's settings, we will be able to examine the tool's full potential in supporting the accessibility features. On the contrary, the second scenario comprises novice web localisers with an insufficient level of knowledge of accessibility, who, therefore, are unaware of possible issues. Through this second scenario, we can determine whether the tools have a positive or negative impact on the degree of accessibility achieved when localisers are not accessibility-savvy. As part of the evaluation design, it was also decided to let participants from both groups make further changes manually in the target file, once exported from the tool, if they deemed it appropriate. By comparing the results of the two groups, more specifically the final target files, we were able to study hypothesis H4.

Ultimately, as we previously mentioned, we wanted to determine the tools' functional appropriateness and, in part, also the functional completeness by reviewing the participants' answers to a post-evaluation questionnaire (see Appendix F). The questionnaire comprised two sections, one per tool, which include the same type of questions. Participants were asked whether they were successful in producing an accessible target file and whether they had to make further manual changes. To have a better understanding of the actual process and to identify possible general issues, participants were also asked to specify for every accessibility

features whether they could localise them or not and if yes, whether they had translated them in the tool's editor interface or if they had manually modified the code. A fourth option ('I don't know') was added in case they were not aware of, and consequently did not consider, that specific accessibility feature.

Furthermore, the researcher deemed it relevant to add one question concerning the information about the feature's context, to determine if participants considered the information offered by the tool useful or not. Another feature that the researcher wanted to investigate is the built-in QA feature: the participants were asked whether the tool showed them any error message and, if yes, whether they were related to any accessibility issue. Ultimately, the last question ('Do you think that using this CAT tool could have a positive or negative impact when localising accessibility features?') was included to collect participants' opinion on this particular issue.

4 Results of the descriptive approach

This chapter provides the results of the descriptive analysis, which we carried out to answer the first research question RQ1. In the first section (4.1), we illustrate the data we collected during the processing of the test webpage with SDL Trados Studio 2017, while the second section (4.2) includes the results we obtained from MemoQ v8.7. The last part (4.3) serve as a recapitulative section: we briefly summarise the findings of the previous sections, which will be later employed for the analysis of the user evaluation (see Chapter 5), and we answer the first research question RQ1 and check hypothesis H1.

Concerning the data analysis approach for the tool descriptive analysis, we decided to adopt a qualitative approach. Although the results collected after the tests on the HTML5 file with both tools were more quantitative, as we were able to count how many accessibility features the tool veritably supported and point out which ones they did not transfer, we wanted to analyse the data from a more qualitative point of view, trying to extract more information from quantitative data (Sandelowski 2000, 253). In other words, we did not limit ourselves to create a list of supported features, but we tried to understand the disadvantages that the eventual failed support and transfer of a specific feature or the advantages of the additional information offered by the tools could have on the achievement of accessibility.

The procedure was as follows: first, we started our analysis by observing the tool's settings concerning HTML that the researcher believed were also related to the selected SC. Second, we ran various tests and manipulated different settings to check all the potential outcomes. We ran four preliminary tests, two per tool, with a different webpage code[15] to test both the default and manipulated settings. Once we selected the definitive test webpage, we ran six additional tests, three per tool. The first two while using default settings, and the following two while using manipulated settings; the final two tests were implemented to verify any eventual doubts, arose during the definition of the study's methodology. We analysed SDL Trados Studio 2017 two more times to examine all the elements listed in the Parser section. The researcher also assessed the overall editor interface to determine whether the tool offered additional

---

[15] The first test webpage selected by the researcher to run the preliminary tests was a webpage on the Swiss Post's website [https://www.post.ch/en/sending-letters/domestic-letters/a-mail, 25.5.2019]. However, as we noticed several problems on the English page, such as elements that were not correctly localised from German, we decided not to use it as our test webpage.

information about the feature's context that could be used by localisers during the web localisation process.

4.1 SDL Trados Studio 2017

4.1.1 General overview of the tool's settings and editor interface

SDL Trados Studio 2017 offers the possibility to modify multiple parameters. During the creation of a new project, users can adapt several elements related to the file type they are working with (see Figure 4.1). These settings can also be modified in later stages, at the end of the creation of the project or by selecting 'Project Settings' in the editor interface. However, we believe it is better to modify them during the creation of a project, as further modifications may not be applied once the project is opened. For instance, during one of the tests, the researcher tried to modify one of the parameters in the settings, yet the tool did not immediately show the attribute among the segments and, consequently, the researcher had to recreate the project.
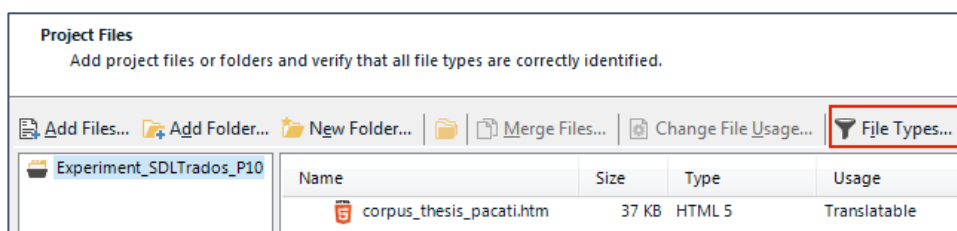
Figure 4.1: Project settings related to the file type.

Concerning the settings that are related to HTML5-format files, there are eight sections in which users can make changes: Detection, Parser, Writer, Entities, Whitespace, Preview, Embedded Content, QuickInsert (see Figure 4.2). Once we observed all the listed sections, we concluded that only Parser and Writer are relevant to our research.
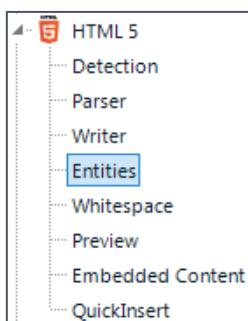
Figure 4.2: Eight sections of the HTML5 settings.

In the latter, users can modify the settings for Unicode UTF-8, the `lang` attribute, and `meta` tags. The second element directly concerns our study, as we included the SC *Language of Page*

53

in our requirements. Users can choose among three options: 'change matching source language to target language', which is the default option; 'always change to target language'; and 'do not change' (see Figure 4.3). In section 4.1.2.5, we will analyse in more details the results of these different options.
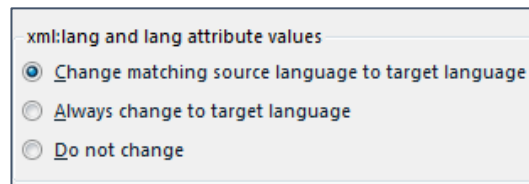
Figure 4.3: Settings related to the `lang` attribute.

The other section that includes important settings related to accessibility is 'Parser' (see Figure 4.4). In this tab, users can either add, edit or remove certain rules associated with embedded elements in the HTML code. When adding or editing a rule, users can either add or edit the rule's attributes and other properties, such as whether or not the tool has to translate it, or whether the tags should be shown in the editable segment, etc. Furthermore, the tool offers the possibility to modify the structure information properties. Among these rules, we noticed several units that can be linked to accessibility, in particular, 'link', 'title', 'a', 'button', 'img', 'input', and 'label'[16]. These rules will be further examined in the corresponding sub-sections.
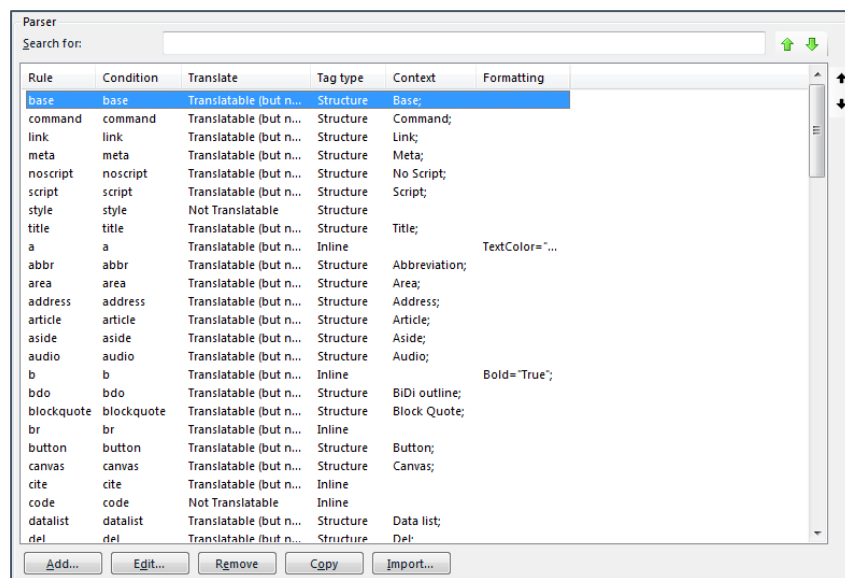
Figure 4.4: Parser tab.

---

[16] The researcher also identified other rules linked to accessibility, such as abbreviations (abbr), input, and tables (table). However, as they were not included in the selected requirements, we did not take them into consideration for our analysis.

Concerning the tool's editor interface, we noticed two important components that users can employ to obtain further information about the feature's context, as indicated in the areas highlighted by two red boxes in Figure 4.5. The red box on the left indicates the information about the document structure. Although we previously stated that we will not focus on the structure of the code, we believe that the information can still be useful to localisers in order to have an idea of which section of the document they are dealing with. Likewise, the second box on the right shows the document structure information tab. Here, we find information about the location of the segment in the document (such as title, attribute, list item, etc.), the code used by the tool to identify it (T+, ATT, LI+, etc.), and other additional information. This type of information can be useful to localisers, who may not be sure about what kind of feature they are dealing with and, consequently, how to translate it.
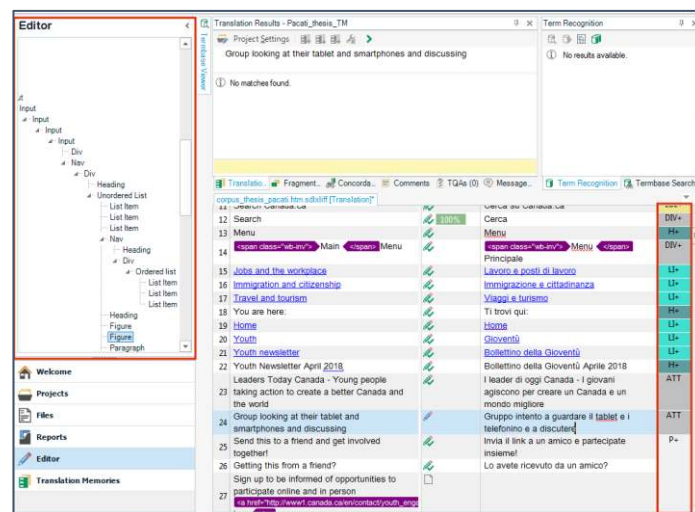
Figure 4.5: SDL Trados Studio 2017's editor interface.

4.1.2 Processing of the SC selected for the study

4.1.2.1 Non-text Content

Concerning the settings related to this SC, users can edit the 'img' rule in the Parser tab. As we are interested in determining whether the tool supports the localisation of alternative texts, we analysed the attributes of the rule (see Figure 4.6) and noticed that the alt attribute is already included as default. Moreover, the translate option is set as 'True', which means that the segment of the text alternative should be shown in the editor.
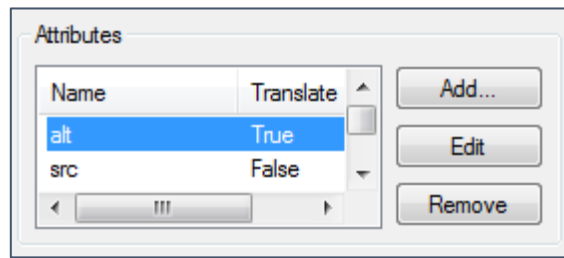
Figure 4.6: `alt` attribute in the 'img' rule.

The segment is indeed shown in the editor interface and therefore localisers can translate it. The tool does not include any additional tag including the protected source code related to the `img` attribute; but it indicates the HTML element and attribute within the segment metadata, as shown in Figure 4.7. However, if localisers do not verify this information in the document structure information, they may not know that they are translating a text alternative.
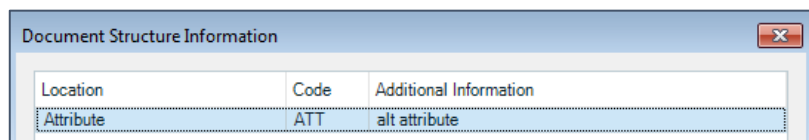
Figure 4.7: Document structure information for the `alt` attribute.

4.1.2.2 Bypass Blocks

In this case, the tool does not offer the possibility to modify any parameter that concerns this particular accessibility feature. Nevertheless, as bypass blocks elements are encoded as a link in the source document, the tool detects them as such. For this reason, the text string is formatted in a different way compared to plain text segments (i.e. underlined and in blue). This visual contrast may prove useful, as localisers can immediately understand what kind of element they are translating. Moreover, in the document structure information tab we noticed that the tool defines the segments as "a group of navigational links" (see Figure 4.8), which indicates that it identifies the feature and, consequently, supports and transfers it correctly.
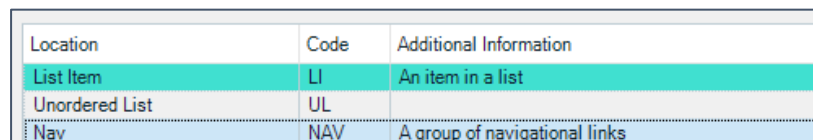
Figure 4.8: Document structure information for bypass blocks feature.

4.1.2.3 Page Titled

As we noticed, the tool includes the `title` attribute among its rules in the Parser section. Therefore, we can state that SDL Trados Studio 2017 supports this feature and, consequently, displays it in the editor. Localisers can obtain additional information through the document structure information tab, where the segment is defined as the title of the document (see Figure

4.9). However, the tool does not display the protected source code inside the segment, and the title is shown as any other plain text unit. Therefore, if users do not check the document structure information tab, they do not have any additional information that can help them identify the title.

Location	Code	Additional Information
Title	T	Title of the document

Figure 4.9: Document structure information for the `title` attribute.

Furthermore, as all the segments with the `title` attribute are automatically detected, the tool also displays several `meta` elements. As there can be useful information embedded in these components, it is worth mentioning that it is possible to show these attributes in the editor. For instance, we can find the page title, the date in which the page has been created and modified, the creator of the page and the language. However, in our case, the translatable content is not included in the `title` attribute, but rather in the `content` attribute. Therefore, users should modify the settings and allow the tool to display this type of information (by adding the `content` attribute to the 'meta' rule in the Parser section). Still, it is worth noting that, by unprotecting it, users may damage the code and modify certain content that should not be localised.

4.1.2.4 Link Purpose (in Context)

In the Parser section, we identified two rules that are related to links: 'link' and 'a'. In our case, we will focus on the latter: the tag type of the `a` rule (anchor element) is already set as 'inline', which means that the tag including the protected source code should be displayed in the editor. However, as we previously mentioned, the tool detects links and displays them in the editor with a different text format, but not all the link segments include tags. As illustrated in Figure 4.10, the tag displays additional information about the protected code that precedes the text unit. Therefore, if tags are not shown, this kind of information is lost.

Figure 4.10: View of a link in the editor interface.

Nevertheless, as for other features, users can find additional information in the document structure information tab. This specific element is indeed recognised as a link, which proves that the tool detects and supports it. Therefore, we can state that, in general, SDL Trados Studio

57

2017 does support links and transfer them correctly, but part of the information about the context may not be displayed.

4.1.2.5 Language of Page

As we illustrated in Section 4.1.1, users can modify settings related to the `lang` attribute in the Writer section. During the first tests, the researcher selected the first option 'change matching source language to target language', which is the set as default, expecting the tool to automatically modify the attribute. However, when analysing the target file, we noticed that the element did not change and the language displayed was still the one corresponding to the source document: English. Therefore, the researcher decided to select the second option ('always change to target language') to verify whether the `lang` attribute could be transferred correctly by the CAT tool. In this case, the attribute did change and the language was correctly set to the target language (i.e. Italian). Hence, we can state that if localisers are not aware of the different outcomes of these two options, and select the first one not knowing that the `lang` attribute will not be modified, there could be an issue and the desired level of accessibility may not be achieved.

4.1.2.6 Language of Parts

Contrary to the SC related to the main language of the page, the tool does not offer any setting concerning the language of parts. In our specific case, there were two elements in French that we localised into a language other than the target language (i.e. in German). However, although the tool displayed the text unit segments in the editor and we could localise the text into another language, the `lang` attribute did not change. Hence, in the target file, the attributes for the French part were still set as such and not in German. This issue could not be resolved in any way (other than manually, after exporting the file); therefore, we can state that the tool does not support this feature and, consequently, we cannot transfer it in the target code.

4.1.2.7 On Input

The settings concerning this SC can be modified in the Parser section. The tag type is set as 'structure' by default, so the researcher tried to change it to 'inline' to see if the tool offered any additional information, but the tags including the source code were not displayed in the segments. Nevertheless, it is still possible to translate the segments, as they include text units that are detected as translatable text strings. Moreover, in the document structure information tab, these elements are described as 'button' (see Figure 4.11). Therefore, we can state that the tool supports and transfers this feature.

Figure 4.11: Document structure information for the `button` attribute.

4.1.2.8 Error Identification

The technique implemented in the test webpage consists of explaining the errors with textual content; hence the tool detects them as translatable elements in the form of plain text units. For this reason, these elements are displayed in the editor as any other translatable textual strings. On the one hand, the fact that errors are treated as plain text units does not cause any issue in the achievement of accessibility; on the other hand, users do not retrieve any additional information concerning this feature (for instance, why it is important) in the document structure information tab.

However, several attributes that preceded the text were displayed in the form of tags, as we can see in Figure 4.12. Among these attributes, we also find `aria-required` attributes, which define whether the field is required or not, and the `id` attribute that specifies the unique id for the HTML element. By showing these tags, localisers can also verify that the right attributes related to accessibility have been embedded correctly in the code.

Figure 4.12: Error identification feature displayed in the editor.

4.1.2.9 Labels or Instructions

As we mentioned in Section 3.2.3.9, we will focus on ARIA attributes. In our test webpage, we identified several ARIA elements, such as `aria-required`, `aria-orientation`, etc. Yet, we found only one element that needed to be localised, namely the `aria-label` attribute included in the `button` element. During the first tests, this element was not displayed by the tool and, consequently, not transferred in the target file. In the second round of tests, the researcher tried to add the attribute to the 'button' rule and set it as translatable (Figure 4.13). As a result, the element was shown in the editor. We can therefore conclude that, although the tool can support and transfer the feature, if users are not aware of this issue or they do not know in which element the ARIA attribute was included in, they may not be able to localise it.

59
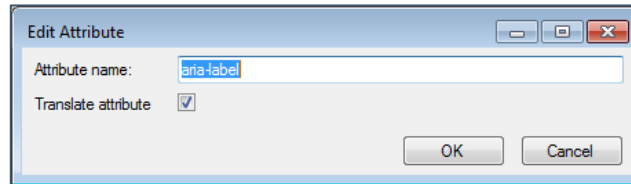
Figure 4.13: `aria-label` attribute added to the 'button' rule.

4.2 MemoQ v8.7

4.2.1 General overview of the tool's settings and editor interface

Compared to SDL Trados Studio 2017, users have the possibility to modify fewer parameters in MemoQ v8.7. When adding the source file, the tool automatically detects the format and applies the corresponding filter. By selecting 'import with options', users can both change the filter (in case the tool did not recognise the right format) and the document import settings by clicking on 'Change filter and configuration' (see Figure 4.14).
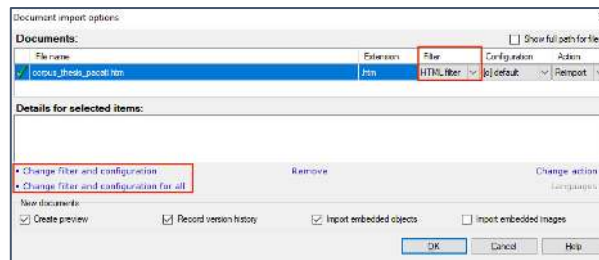
Figure 4.14: Filter and configuration in MemoQ v8.7.

In the new tab (see Figure 4.15), users can select the import type: import markup as either inline tags (which is set as default) or with MemoQ's tags. For the purpose of our research, we decided to keep the first option to check which part of the protected code the tool showed in the editor. Moreover, users can choose the import settings, such as the HTML configuration and the PHP (Personal Home Page) configuration. Concerning the former, it is possible to select or unselect the following settings: ignore content of drop-down lists; import non-breaking spaces as entities; enforce empty `<img/>` and `<br/>` tags; treat `<br/>` tags as inline; import access keys; import HTML entities as characters; always use HTML entities in export; break segment at preserved newline characters; and import processing instructions as inline tags. For the first round of tests, the researcher chose to keep the default settings when processing the test webpage's code, as some of the parameters we believed were relevant to our research were already selected and others did not concern directly the requirements.
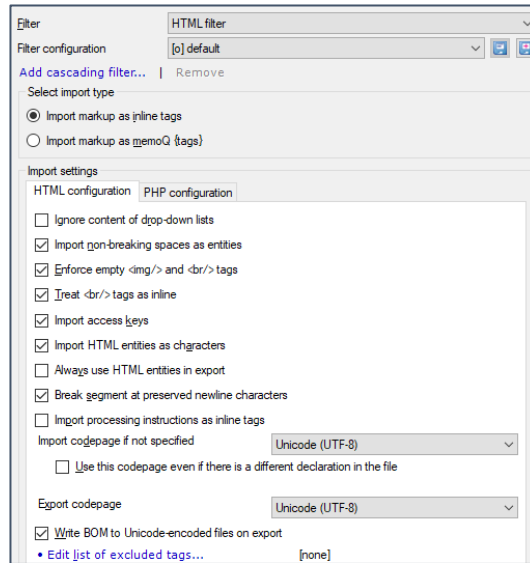
Figure 4.15: Document import settings in MemoQ v8.7.

Other settings can be modified in the project home, such as the segmentation rules, the QA settings, or the auto-translation rules. However, we did not find any parameter concerning the selected SC or the format file type. Overall, users are able to manipulate only a small number of settings, of which only a few are related to the accessibility features we are interested in.

The editor interface presents, in addition to the Grid and the translation results pane, the view pane where users can see a formatted preview of the document, shown inside a red box in Figure 4.16. The tool highlights the segment that the localiser is translating and, once the user confirms the translation, the segment changes automatically. As we will see in the following sub-sections, this functionality may be useful while localising certain accessibility features. This option is also available in SDL Trados Studio 2017, but it is not fully integrated in the editor interface.
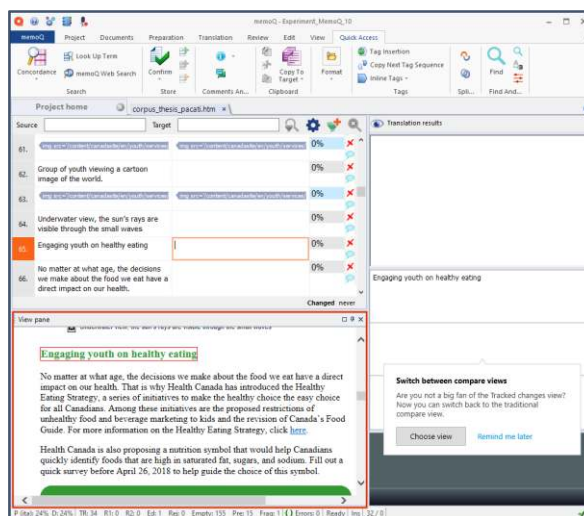
Figure 4.16: MemoQ v8.7's editor interface.

61

Another aspect that we deem relevant to highlight is the built-in QA function. SDL Trados Studio 2017 also offers this feature (QA Checker 3.0), but it only checks punctuation, numbers, and regular expressions, while MemoQ v8.7 is also able to check inline tags. Before finalising the project, the tool detects any error or missing element in the target file, including issues related to accessibility. In the following sub-sections, we will also illustrate some of the errors detected by MemoQ v8.7.

4.2.2 Processing of the SC selected for the study

4.2.2.1 Non-text Content

As we mentioned in the previous section, the tool presents a setting related to images ('enforce empty `<img/>` and `<br/>` tags'), but we do not find any setting that concerns exclusively text alternatives. Nevertheless, although users are not able to manipulate this feature, the tool still supports it and shows the `alt` attribute in the editor, which means that it is set as translatable by default. As we can see in Figure 4.17, there is additional information in the previous segment, in which the original code containing the `img` element that precedes the `alt` attribute segment is shown. This aspect can be useful during the localisation process, as users are aware of the element they are translating.
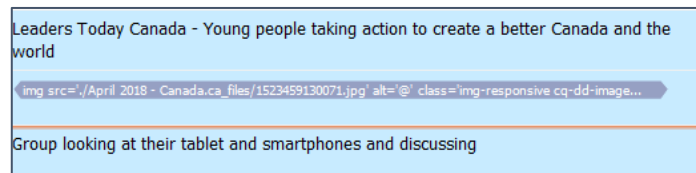
Figure 4.17: Segments including the `alt` attribute and 'img' element.

Another function that may have a positive impact on the localisation of the `alt` attribute is the view pane. Here, as illustrated in Figure 4.18, users can both view the image (in our case, as we did not include the images as reference, we can only see a symbol representing the image) and the text alternative. Thus, by looking at the view pane, users can quickly verify which element they are translating and localising it accordingly.
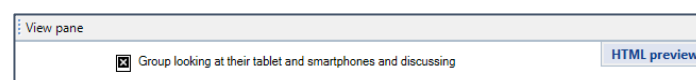
Figure 4.18: `alt` attribute preview.

Moreover, the built-in QA function displays error messages concerning the `alt` attribute, as illustrated in Figure 4.19. In this case, the expression 'contains direct text' indicates that the value introduced in the translatable attribute is incorrect. This function may be an additional instrument that localisers can use to improve the localisation of text alternatives.

Figure 4.19: Error message related to the alt attribute.

4.2.2.2 Bypass Blocks

Concerning the settings, the tool does not offer the possibility to modify directly this specific element. However, although we did not take this aspect into consideration, the tool presents the option to import 'access keys'. Access keys are keyboard shortcuts, which can be employed to navigate the page. They may be neutrally transferred during the localisation process, but by showing them in the editor, localisers can verify whether they have been implemented correctly. Similarly to SDL Trados Studio 2017, as bypass blocks elements are embedded in the code as links, the tool identifies them as such. For this reason, as we can see in Figure 4.20, the tool detects both the translatable text units and shows the markup content that precedes the text. By showing the source code, the tool offers more context about the feature.

Figure 4.20: Bypass blocks elements.

Although these elements are not visible in the actual webpage, as they are only detected by ATs, it is possible to see them in the view pane (see Figure 4.21). This is a positive factor, as it proves that the tool can identify elements that are embedded in the code and that are set as invisible in the webpage. This aspect is missing from SDL Trados Studio 2017, as the preview tab displays the final product as it would be presented to end-users.
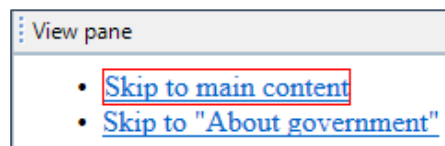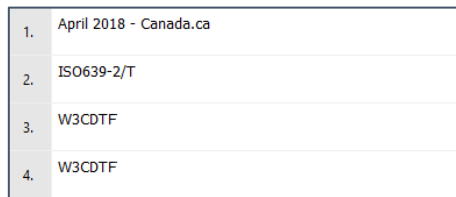
Figure 4.21: Bypass blocks in the view pane.

4.2.2.3 Page Titled

Similarly to SDL Trados Studio 2017, MemoQ v8.7 identifies the `title` attribute and displays it correctly in the editable segment, despite the fact that there are no particular parameters that concern this specific feature. However, as in the case of the first tool we reviewed, it also detects the attribute included in the `meta` elements (see Figure 4.22). This information remains unprotected by default and it is up to localisers to report the content without making any

modification. Contrary to SDL Trados Studio 2017, users do not have the possibility to display the `content` attribute, in which we can find translatable elements.

Figure 4.22: `title` attribute and metadata elements.

4.2.2.4 Link Purpose (in Context)

As we have previously illustrated in Section 4.2.2.2, links are displayed with both the protected source code and the translatable text unit (see Figure 4.23). Contrary to SDL Trados Studio 2017, MemoQ v8.7 always includes the tags, which display the markup content, in the editable link segment. By setting the tag length to 'long', users can view the `href` attribute of the anchor element, which shows the URL of the actual link, and, consequently, can offer more context about the feature. This aspect can have a positive impact on the localisation of links, as users are aware of the link path and the content of the page it redirects to.
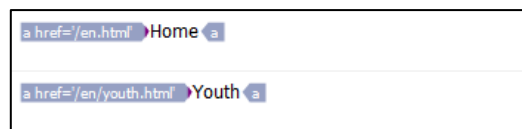
Figure 4.23: Anchor elements.

In the view pane, users can easily identify links as they have a different format. Like in the first CAT tool reviewed, links are underlined and in blue. However, the different format is visible only in the view pane and not the translatable segments.
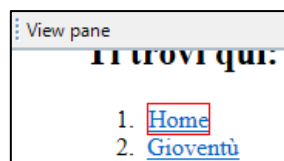
Figure 4.24: Links preview in the view pane.

4.2.2.5 Language of Page

Contrary to SDL Trados Studio 2017, MemoQ does not offer the possibility to modify any settings related to the language of the document. During the first tests run by the researcher, the tool automatically changed the `lang` attribute to the target language, from English to Italian. This led us to believe that the tool supported and could transfer this feature. However, once we analysed the target files produced by the participants during the tool evaluation, we noticed that

in all cases the attribute did not change. Therefore, to clarify this aspect, we run additional tests and came to the conclusion that the support of this feature depends on the language pair selected during the creation of the project. At first, the researcher selected English and Italian as the language pair, without specifying the variety of the two locales; while during the user evaluation, participants selected English (en-CA) and Italian (it-IT). In the source code, the `lang` attribute has only 'en' as the set value, and therefore the tool identifies it only as 'English' and modifies it accordingly. This is a fundamental aspect, because if localisers do not check which variant of source language is included in the language attribute, the feature may not be transferred correctly by the tool.

4.2.2.6 Language of Parts

As in SDL Trados Studio 2017, there are no settings related to this SC. The two translatable text units in French are retrieved and shown in the editor and we tried to adapt them in German as we did for the first tool; however, this did not prove successful and, although the text was translated correctly, the `lang` attribute did not change. We can therefore conclude that neither MemoQ supports this feature, nor can transfer it correctly.

4.2.2.7 On Input

Concerning the settings, the tool does not offer the possibility to modify this element. However, contrary to the first CAT tool we examined, MemoQ does include the source code in the segments that concern this feature. As it identifies translatable text units, localisers can adapt submit buttons. Through the markup language displayed in the tags that accompany the text, users can verify the purpose of the button and retrieve additional information about the element. This factor has a positive aspect on the localisation of submit buttons, as users are informed about the element and can therefore adapt it correctly.

4.2.2.8 Error Identification

As we mentioned in Section 4.1.2.8, the technique implemented in the source code includes text units that are not embedded in the code. As a result, the tool identifies them as plain text units and displays them in the editor. As we can see in Figure 4.25, MemoQ also includes tags that display the source code, which consists in the anchor element. However, the details comprised in the tag do not offer useful information related to the errors messages, as there is only the `href` attribute. The `aria-required` attribute and the `id`, which we previously mentioned, are not displayed in the tag. Therefore, we can state that MemoQ supports and transfers the error messages, as they are identified as plain textual units, but it does not offer any additional information related to the feature's context.

Figure 4.25: Error identification elements.

4.2.2.9 Labels or Instructions

Concerning this last feature, we firstly analysed whether the tool presented any settings related to ARIA elements, with negative results. Moreover, as the `aria-label` attribute is embedded in the code and included in the button attribute, the tool does not retrieve this feature and, consequently, does not display it in the editor. The feature is also not visible in the view pane, which lead us to believe that the tool does not support this feature and, therefore, is not able to transfer it.

4.3 Tool descriptive analysis: summary of the main findings

In this last section, we briefly summarise the data collected in order to answer to the first researcher question RQ1 ('Can CAT tools support and therefore transfer all the relevant accessibility features when processing an HTML5 file?'), and check the first hypothesis H1 ('CAT tools do not support and transfer all the accessibility features.'). In the following table (Table 4.1), we show the results for both SDL Trados Studio 2017 and MemoQ v8.7. Findings are grouped in the following categories: settings manipulation, the actual support and transfer of the feature, and the eventual additional information provided by the tools. In green, we indicated the elements offered by the tools, while in red we showed the missing features and functionalities.

66

	SDL Trados				MemoQ v8.7			
	Settings	Support	Transfer	Additional information	Settings	Support	Transfer	Additional information
Non-text Content	G	G	G	G	G	G	G	G
Bypass Blocks	R	G	G	G	G	G	G	G
Page Titled	G	G	G	G	R	G	G	R
Link Purpose	G	G	G	G	R	G	G	G
Language of Page	G	G	G	R	R	G	G	R
Language of Parts	R	R	R	R	R	R	R	R
On Input	G	G	G	G	R	G	G	G
Error Identification	R	G	G	G	R	G	G	R
Labels or Instructions	G	G	G	R	R	R	R	R

Table 4.1: Summary of the results of the tool descriptive analysis.

At a glance, we notice that SDL Trados Studio 2017 supports and transfers a higher number of features compared to MemoQ v8.7. However, we deem it important to highlight the fact that some of these elements may not be transferred if localisers do not modify certain settings or are not aware of the outcome of the specific parameters offered by the tool. This is the case, for instance, of SC such as *Language of Page* and *Labels or Instructions*.

In general, we can state that both CAT tools retrieve and detect correctly the translatable elements that are not embedded in the code. For instance, both systems displayed in the editor interface the title of the page, the links and the error messages. Concerning the first element, both tools also included a few segments that contained content that should not be translated, i.e. the value of `title` attributes included in the `meta` element. It is important to highlight this issue, as the code could be damaged if localisers do not neutrally transfer these elements. In relation to links, SDL Trados Studio 2017 offered less useful information compared to MemoQ v8.7, as in several segments, the source code, which can provide information about the link path, was not included. Moreover, the markup content displayed in the error messages segments in MemoQ v8.7 did not provide any context about the errors.

Concerning the embedded content, we collected mixed data. On the one hand, both tools present settings related to text alternatives, can support and transfer `alt` attributes, and also provide additional information about the feature's context. This is a positive aspect, as the localisation

of non-text content is an important step in achieving accessibility. On the other hand, we identified several issues related to the `lang` attribute. Although both tools can support and adapt the target language of the document, users must be aware of the different settings and the selection of the language pair. In SDL Trados Studio 2017, users could be deceived by the option concerning this element, while in MemoQ v8.7 users should check the exact locale of the source language and select the language pair accordingly. Moreover, both tools were not able to transfer the language of parts, which is the only problem we could not resolve for the two CAT tools. Another issue that the researcher could not overcome was the localisation of the `aria-label` attribute in MemoQ v8.7. While in SDL Trados Studio 2017 it is possible to add this element, in MemoQ v8.7 it appears that users cannot modify the settings.

Overall, SDL Trados Studio 2017 offers the possibility to change more elements in the settings compared to MemoQ v8.7, which has import settings related to only *Non-text Content* and *Bypass Blocks*. Although we can state that, from this point of view, SDL Trados Studio 2017 is more flexible, the results we obtained with both tools were similar. We could hypothesise, therefore, that having knowledge of accessibility can be decisive when using both tools.

Furthermore, in relation to additional information, we noticed that SDL Trados Studio 2017 provides more information, but users should be familiar with the tool, as some of this information is not directly displayed in the editor, but rather in the document structure information tab. In MemoQ, on the contrary, the additional information was primarily included in the form of tags that showed and protected the original source code.

With the tool descriptive analysis of the two CAT tools, we could confirm the first hypothesis H1, which stated that CAT tools do not support and transfer all the accessibility features. Nevertheless, the outcome was better than what we had envisaged: SDL Trados Studio 2017 could not transfer only one feature, while MemoQ v8.7 could not transfer two. Therefore, we cannot answer affirmatively or negatively to the first research question RQ1. On the one hand, both tools seem to support the transfer of the majority of the selected SC; on the other hand, we still need to take into consideration that not all accessibility features could be transferred, and that many variables can have an impact on the transferability of these elements, such as the knowledge about the tool and the experience of the user, the processed file type and accessibility. All these aspects have been considered in the second stage of our research.

5 Results of the tool evaluation

This chapter provides the results for the experimental study. The data analysed include the following: the target files produced by novice localisers (version 1 and version 2), the participants' answers to a post-evaluation questionnaire and the notes taken by the researcher through the observation of participants during the experiment. These data will be examined to answer the second research question RQ2 and check hypothesis H2, H3, and H4. In Section 5.1, we measure the effect that the first independent variable (CAT tools) caused on the three dependent variables, namely *functional correctness*, *functional completeness*, and *functional appropriateness*. Section 5.2 deals with the second independent variable (knowledge of accessibility) to examine the last hypothesis H4. Last, in Section 5.3, we summarise the main findings and discuss them to draw some preliminary conclusions.

5.1 Functional suitability

As we explained in Chapter 3, the second stage of our research consisted of determining the functional suitability of both SDL Trados Studio 2017 and MemoQ v8.7. To do so, we manipulated the two CAT tools to measure the three sub-characteristics of *functional suitability*: *functional correctness*, *functional completeness*, and *functional appropriateness*.

To measure the first sub-characteristic, *functional correctness*, we analysed the target files produced by the participants using the two CAT tools (version 1). As we mentioned in Section 3.4.3, we employed the selected SC as 'measurable attributes' (see Table 3.3); therefore, through the analysis of the target files and the quantitative data collected, we calculated the degree of accessibility achieved by every participant. This was done by checking if all the accessibility features included in the code were transferred correctly. For instance, the `alt` text for non-text content appears 18 times in the code: to calculate this element, we examined whether the element was localised each time and counted eventual errors. We then took into consideration these results to answer the second research question RQ2 ('Is the resulting file accessible?').

To assess *functional completeness*, we considered the selected SC as tasks that participants had to accomplish. By calculating the success rate for every participant, we determined whether the tools offered sufficient functions to cover all the specified tasks. The results also determined the impact that functional completeness has on the final achievement of accessibility. All this

allowed us to check hypothesis H2 ('The functional completeness of the tool used has an impact on the final degree of accessibility achieved').

Ultimately, to measure the last sub-characteristic, *functional appropriateness*, we adopted a mixed approach, by retrieving both quantitative and qualitative data, as we analysed the answers to a number of questions included in the post-evaluation questionnaire. By examining their answers, we could check hypothesis H3 ('The functional appropriateness of the tool used has an impact on the final degree of accessibility achieved').

As we presented in Chapter 3, we divided participants into two groups (see Table 3.5): one with basic knowledge of accessibility and experience with accessibility issues (Group 1), and one with no experience with accessibility issues and no or basic knowledge of accessibility (Group 2). Therefore, we took into consideration this aspect during the analysis of the findings, to highlight possible differences between the two groups.

5.1.1 Functional correctness

In the methodology chapter (Chapter 3), we described *functional correctness* as the degree to which the system can produce an accessible target file. To measure this aspect, as we introduced in the previous section, we calculated the percentage of accessibility achieved by every participant in the target file they exported from the tool (without any manual modification). This was done by examining all the selected accessibility features that needed to be localised (80 in total) and checking if they were transferred correctly. In other words, we counted any eventual errors and, subsequently, calculated the percentage. In the following sub-sections, we present the results for SDL Trados Studio 2017 and MemoQ v8.7, respectively, and we summarise the main findings.

5.1.1.1 SDL Trados Studio 2017

When asked whether they were able to produce an accessible target file, all participants (N=10) answered affirmatively to the question. However, as we can see from the results in Table 5.1, there were several accessibility features that were not transferred correctly in the target product: the language attribute for both the main language of the document and the parts in French and the SC concerning the `aria-label` attribute. In the following table, we indicate the total number of occurrences for each SC (highlighted in grey) and the number of elements per SC that the participants correctly localised in the target files (out of 80 measurable attributes). We divided the participants into groups (Group 1 in blue and Group 2 in orange), and in green, we

highlighted the total of localised elements per participants, the mean, and the standard deviation (SD).

SC	N° of total occurrences	GROUP 1					GROUP 2					MEAN	SD
		P03	P06	P07	P08	P10	P01	P02	P04	P05	P09		
1.1.1 Non-text Content	18	18	18	18	18	18	18	18	18	18	18	18	0
2.4.1 Bypass Blocks	3	3	3	3	3	3	3	3	3	3	3	3	0
2.4.2 Page Titled	1	1	1	1	1	1	1	1	1	1	1	1	0
2.4.4 Link Purpose (in Context)	46	46	46	46	46	46	46	46	46	46	46	46	0
3.1.1 Language of Page	1	1	0	0	0	1	0	0	0	0	0	0.2	0.42
3.1.2 Language of Parts	2	0	0	0	0	0	0	0	0	0	0	0	0
3.2.2 On Input	3	3	3	3	3	3	3	3	3	3	3	3	0
3.3.1 Error Identification	5	5	5	5	5	5	5	5	5	5	5	5	0
3.3.2 Labels or Instructions	1	0	0	0	0	1	0	0	0	0	0	0.1	0.32
TOTAL	**80**	77	76	76	76	78	76	76	76	76	76	76.3	0.67

Table 5.1: Accessibility features transferred with SDL Trados Studio 2017.

We can see that a few participants of Group 1 (P03 and P10) obtained better results compared to the others (N=8), as they were able to transfer more elements: P03 could transfer 77 elements out of 80, while P10 transferred 78 accessibility features. More specifically, both participants modified the settings related to the `lang` attribute by selecting the option 'Always change to target language', and P10 decided to add the `aria-label` attribute to the 'input' rule in order to display the element. The other participants in Group 1 (N=3) could transfer 76 features out 80, as they did not change any parameters in the settings.

Similarly, all the participants in Group 2 (N=5) decided to stick with the default settings and did not manipulate any parameter. As a result, they were able to transfer 76 elements out of 80. The accessibility features that were not localised correctly in the target files were SC *Language of Page*, *Language of Parts*, and *Labels or Instructions*.

In general, all participants (N=10) could transfer more than half of the selected requirements, more specifically the SC concerning alternative texts, bypass blocks, the title of the page, links, buttons, and error messages. We had already concluded, thanks to the results from the tool descriptive analysis, that SDL Trados Studio 2017 cannot support and transfer the language attribute for the parts in French, and, for this reason, we did not expect participants to succeed. In fact, none of them accomplished this task.

On the contrary, we verified that the tool can support the SC *Language of Page* and *Labels or Instructions*. Yet, almost all participants (N=8) did not modify the settings concerning the

language attribute for the whole document and chose to stick with the default option ('Change matching source language to target language'). Also, the majority (N=9) did not add the specific attribute related to the `aria-label` element. As a result, these two elements were still in English in nearly all target files and the final degree of accessibility was inferior compared to the highest attainable level of accessibility allowed by the tool.

Once we collected all the results illustrated above, we decided to calculate the degree of accessibility achieved by every participant, as an overall percentage (see Figure 5.1). We notice that the average level of accessibility achieved in Group 1 (95.7%, SD=0.9798) is slightly higher than the score obtained by Group 2 (95%, SD=0). In general, data indicated that the average level of accessibility achieved by all participants is 95.4% (SD=0.7762). These results support the findings of the tool descriptive analysis, which showed that the tool supported and transferred the majority of the selected requirements. It also highlights the issues we pointed out in the previous chapter, related to the `lang` and `aria-label` attributes, as almost all participants did not modify the setting accordingly and, in consequence, did not transfer these specific accessibility features.
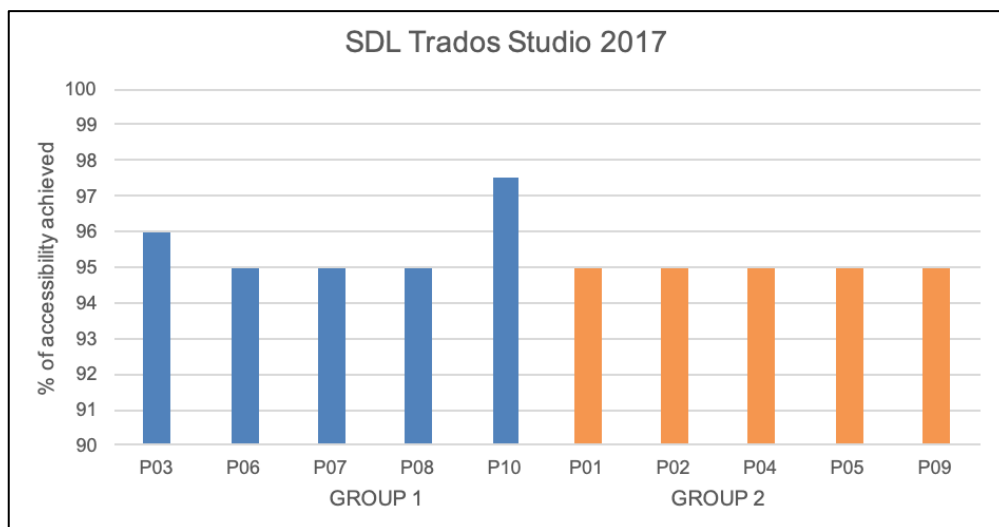
Figure 5.1: Degree of accessibility (in %) achieved by every participant.

5.1.1.2 MemoQ v8.7

As in the case of SDL Trados Studio 2017, all participants (N=10) gave an affirmative answer when asked whether they were able to produce an accessible product with MemoQ v8.7. However, as in the case of the first tool we reviewed, we noticed several issues during the analysis of the target files. In Table 5.2, we indicate the number of total occurrences for each SC and all the elements per SC that the participants were able to transfer using MemoQ v8.7. In yellow, we highlighted the total, the mean, and the SD. The first thing we can observe is that

all participants (N=10) produced the same results and transferred a total of 76 accessibility features out of 80. Therefore, we can state that the mean percentage of accessibility achieved in the target files by all participants is 95% (SD=0).

SC	N° of total occurrences	GROUP 1					GROUP 2					MEAN	SD
		P03	P06	P07	P08	P10	P01	P02	P04	P05	P09		
1.1.1 Non-text Content	18	18	18	18	18	18	18	18	18	18	18	18	0
2.4.1 Bypass Blocks	3	3	3	3	3	3	3	3	3	3	3	3	0
2.4.2 Page Titled	1	1	1	1	1	1	1	1	1	1	1	1	0
2.4.4 Link Purpose (in Context)	46	46	46	46	46	46	46	46	46	46	46	46	0
3.1.1 Language of Page	1	0	0	0	0	0	0	0	0	0	0	0	0
3.1.2 Language of Parts	2	0	0	0	0	0	0	0	0	0	0	0	0
3.2.2 On Input	3	3	3	3	3	3	3	3	3	3	3	3	0
3.3.1 Error Identification	5	5	5	5	5	5	5	5	5	5	5	5	0
3.3.2 Labels or Instructions	1	0	0	0	0	0	0	0	0	0	0	0	0
TOTAL	**80**	76	76	76	76	76	76	76	76	76	76	76	0

Table 5.2: Accessibility features transferred with MemoQ v8.7.

This aspect can be explained by reviewing the findings of the descriptive approach. We already pointed out that MemoQ v8.7 cannot support and transfer two elements, namely SC *Language of Parts* and *Labels or Instructions*. Therefore, we did not expect participants to accomplish these two tasks in the first place. As in the case of SDL Trados Studio 2017, participants could not transfer these features automatically through the tool. If we compare the results of both tools, we notice that the two participants who transferred more elements (P03 and P10), in this case, produced the same target files as the other participants. Consequently, the two groups obtained the same results. This aspect supports the findings of the descriptive approach: the tool does not offer the possibility to modify the settings related to most SCs and, consequently, all participants did not manipulate any of the parameters and worked with the default settings. Various reasons may have influenced the participants' decision to not change the settings: time constraints, the participants' limited knowledge of the tool, or the lack of knowledge of accessibility. The latter, however, does not apply to participants P03 and P10, as they did modify several parameters with SDL Trados Studio 2017, but did not do the same for MemoQ v8.7. This may be due to the fact that they did not find any parameter related to the feature they wanted to transfer or to the fact they believed that the default settings were sufficient. Yet, although said participants had sufficient knowledge of accessibility to improve the degree of accessibility achieved using SDL Trados Studio 2017, they could not do the same using

MemoQ v8.7. This aspect in part proves that MemoQ v8.7 is less flexible compared to the first tool we reviewed.

Concerning the language attribute for the whole document, the researcher instructed the participants to set the source language as en-CA, and not just English, when selecting the project language pair, which, in consequence, caused the missing support and transfer of the correct `lang` attribute. This undesired bias involuntarily introduced by the researcher part, however, led us to discover this issue related to the language of the document.

5.1.1.3 Summary of the findings

In this last sub-section, we briefly summarise the findings that will allow us to answer the second research question RQ2. As we illustrated above, the average percentage of accessibility achieved by the ten participants using SDL Trados Studio 2017 (95.4%, SD=0.7762) is slightly higher compared to MemoQ v8.7 (95%, SD=0). The difference lies in the fact that two participants of Group 1 could modify the settings in the first tool and, consequently, improve the support of certain elements, but they could not do the same thing for the second tool. However, the results are almost the same and confirm the findings illustrated in the previous chapter: participants could transfer the majority of accessibility features, but none of them could transfer all of the elements. These results are due to both the tools limited support of certain SC (SC *Language of Parts* in the case of SDL Trados Studio 2017 and SC *Language of Parts* and *Labels or Instructions* in the case of MemoQ v8.7), and the decision made by participants regarding the manipulation of the tools' settings.

We also need to take into consideration that the high percentage of accessibility achieved is related to the number of occurrences of SC, as links count for more than half of the measurable attributes, and another 20% consists of alternative texts, while the `lang` attribute appears only three times in the whole code. Therefore, in order to answer the research question, we need to take into account the fact that both tools did not transfer 100% of SC, for the reason we mentioned in the previous paragraph.

All the above considered, we need to answer negatively to the second research question RQ2, which investigates whether the resulting file exported from the CAT tool is fully accessible, as none of the participants transferred all the selected accessibility features.

5.1.2 Functional completeness

In the context of our research, we can define *functional completeness* as the degree to which the tool's feature and functionalities cover all the tasks. In other words, we can measure the

system's capacity to provide the right functions specified by the user to attain the desired result. As we explained in Chapter 3, we divided the selected SC into nine tasks that participants had to accomplish to obtain an accessible target file.

5.1.2.1 Participants' awareness of the selected SC

In this section, we review some of the answers to the post-evaluation questionnaire, which will help us interpret and discuss the success rate scores. In the questionnaire, participants were asked to indicate, SC by SC, whether they could automatically transfer the elements through the CAT tool, and whether they had to make manual modifications. A fourth option, 'I don't know', was provided in case participants were not aware of some of the selected features (see questions A3-A11 and B3-B11 in Appendix F). In Table 5.3, we indicate the answers concerning SDL Trados Studio 2017[17]. The colour yellow indicates the answer 'Transferred through the tool', blue indicates 'Through manual modification', while red shows the negative answers ('No'), and green shows the answers 'I don't know'. We also included a summary for each participant, in which we indicate the number for each selected answer.

SC	GROUP 1					GROUP 2				
	P03	P06	P07	P08	P10	P01	P02	P04	P05	P09
1.1.1 Non-text Content										
2.4.1 Bypass Blocks										
2.4.2 Page Titled										
2.4.4 Link Purpose (in Context)										
3.1.1 Language of Page										
3.1.2 Language of Parts										
3.2.2 On Input										
3.3.1 Error Identification										
3.3.2 Labels or Instructions										
Through the tool	6	4	4	7	6	4	8	5	7	6
Manual modification	2	5	2	2	3	0	0	1	0	0
No	1	0	0	0	0	3	1	0	1	1
I don't know	0	0	3	0	0	2	0	3	1	2

Table 5.3: Answers related to the transfer of accessibility features in SDL Trados Studio 2017.

We noticed that most participants included in Group 2 – *No accessibility knowledge* (N=4) selected 'I don't know' to answer at least one question, which shows that they were not aware of one or more elements included in the selected requirements that they needed to localise. On

---

[17] These answers do not reflect necessarily the results we obtained through the analysis of the target and final files, but they indicate what the participants thought. This aspect will be discussed in Section 5.2.

the other hand, the majority of participants in Group 1 – *Basic accessibility knowledge* (N=4) detected all the selected accessibility features. Only one participant in Group 1 (P07) indicated that they did not know about three elements (SC *Bypass Blocks*, *Page Titled*, and *Labels or Instructions*).

Table 5.4 shows the results for MemoQ v8.7. Again, we can observe that the same participants selected the fourth option ('I don't know') to indicate that they did not detect certain elements.

SC	GROUP 1					GROUP 2				
	P03	P06	P07	P08	P10	P01	P02	P04	P05	P09
1.1.1 Non-text Content										
2.4.1 Bypass Blocks										
2.4.2 Page Titled										
2.4.4 Link Purpose (in Context)										
3.1.1 Language of Page										
3.1.2 Language of Parts										
3.2.2 On Input										
3.3.1 Error Identification										
3.3.2 Labels or Instructions										
Through the tool	6	5	4	6	6	6	8	5	7	7
Manual modification	2	4	2	3	3	0	0	1	0	0
No	1	0	0	0	0	2	1	0	0	1
I don't know	0	0	3	0	0	1	0	3	2	2

Table 5.4: Answers related to the transfer of accessibility features in MemoQ v8.7.

These answers show that half of the participants (N=5) was not familiar with one or more elements included in the selected SC and, therefore, could not verify whether they were transferred correctly.

5.1.2.2 Success rate of both CAT tools

Concerning the success rate for SDL Trados Studio 2017, as illustrated in Table 5.5, six tasks were accomplished by all the participants (N=10), while Task 6 (SC *Language of Parts*) could not be completed by anyone. We notice that the average success rate score of Group 1, 66% (SD=8), is higher compared to the score of Group 2, 60% (SD=0), as P03 completed Task 5 (SC *Language of Page*) and P10 could accomplish both Task 5 and Task 9 (SC *Labels or Instructions*).

| SC | GROUP 1 | | | | | GROUP 2 | | | | | % |
	P03	P06	P07	P08	P10	P01	P02	P04	P05	P09	
TASK 1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 4	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 5	Yes	No	No	No	Yes	No	No	No	No	No	20
TASK 6	No	No	No	No	No	No	No	No	No	No	0
TASK 7	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 8	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 9	No	No	No	No	Yes	No	No	No	No	No	10
%	70	60	60	60	80	60	60	60	60	60	63

Table 5.5: Success rate per task per participant (SDL Trados Studio 2017).

As we previously mentioned, SDL Trados Studio 2017 offers the possibility to modify the settings related to the language and `aria-label` attributes, which, if changed correctly, can improve the tool's performance. Among participants of Group 1, only one person (P07) indicated that they were not sure about three specific elements SC *Bypass Blocks* (Task 2), *Page Titled* (Task 3), and *Labels or Instructions* (Task 9). However, the results show that they could accomplish both Task 2 and Task 3. The only task the participant could not complete because of the lack of knowledge of accessibility was Task 9.

In the case of Group 2, four participants (P01, P04, P05, and P09) indicated that they did not know whether they could localise several elements. Among these features, we find *Bypass Blocks* (Task 2), *Page Titled* (Task 3), *Language of Page* (Task 5), *On Input* (Task 7), *Error Identification* (Task 8), and *Labels or Instructions* (Task 9). However, in the case of Task 2, Task 3, and Task 7, all participants from Group 2 (N=5) transferred correctly all the said accessibility features. On the other hand, none of them accomplished Task 5 and Task 9 for the reasons previously mentioned.

This factor demonstrates that SDL Trados Studio 2017 can have a positive impact on the localisation of certain elements (in our case of SC *Bypass Blocks*, *Page Titled*, *On input*, and *Error Identification*): although users may not be aware of certain elements they need to localise, the tool supports and transfers them by default. On the contrary, concerning Task 5 and Task 9, if localisers are not aware of these issues and do not modify the corresponding settings, the tool does not help them in localising these elements. This aspect is supported by the findings,

as a few participants (N=2) of the group with basic knowledge of accessibility and experience with accessibility issues (Group 1) detected the elements they had to modify and changed the settings accordingly, which led to a higher success rate score. In contrast, participants with no experience with accessibility issues did not have sufficient knowledge to modify the corresponding parameters and produced a target file with an inferior degree of accessibility conformance. We deem it relevant to highlight this aspect, as it introduces an important argument that we will take into consideration during the discussion in Section 5.2.

Moreover, we also calculated the success rate for MemoQ v8.7. As we previously pointed out, all the participants (N=10) produced the same results and, therefore, the success rate is 60% (SD=0). All participants completed six tasks but did not accomplish the remaining three (Task 4, Task 5, and Task 9).

SC	GROUP 1					GROUP 2					%
	P03	P06	P07	P08	P10	P01	P02	P04	P05	P09	
TASK 1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 4	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 5	No	No	No	No	No	No	No	No	No	No	0
TASK 6	No	No	No	No	No	No	No	No	No	No	0
TASK 7	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 8	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	100
TASK 9	No	No	No	No	No	No	No	No	No	No	0
%	60	60	60	60	60	60	60	60	60	60	60

Table 5.6: Success rate per task per participant (MemoQ v8.7).

As in the case of the first CAT tool, one participant in Group 1 (P07) mentioned SC *Bypass Blocks*, *Page Titled*, and *Language of Parts* as the elements they were not aware of. The tool allowed them to transfer correctly the first two SC but did not support the last. As for Group 2, participants (N=4) indicated *Bypass Blocks* (Task 2), *Page Titled* (Task 3), *Language of Page* (Task 5), *Error Identification* (Task 8), and *Labels or Instructions* (Task 9), together with *Language of Parts* (Task 6), as SC they were not sure of. Even in this case, we noticed that the tool helps localisers transfer the elements included in Task 2, Task 3, and Task 8, as the default settings already implement these accessibility features. On the other hand, the tool hinders the correct transfer of elements included in Task 6 and Task 9, as the tool does not support them,

and may also have a negative impact on the localisation of Task 5 if users do not apply the right settings.

Although the majority of participants in Group 1 (N=4) was aware of all the selected requirements, their success rate was the same as Group 2. As we explained in the section related to functional correctness (Section 5.1.1.2), this is caused by the fact that all participants (N=10) worked with default settings. Therefore, we can state that MemoQ v8.7 does not offer features and functionalities that facilitate the accomplishment of certain tasks and, although localisers may have sufficient knowledge to deal with accessibility issues, the tool's limited performance hinders the achievement of accessibility.

5.1.2.3 Summary of the findings

In line with the findings presented in Section 5.1.1.3, we can state that the success rate of SDL Trados Studio 2017 (63%, SD=8) is slightly higher than the success rate obtained through MemoQ v8.7 (60%, SD=0). We noticed that participants completed the majority of tasks, but did not accomplish several others (Task 5, Task 6, and Task 9) because of both the limited support offered by the two tools and the decision of the participants to not modify certain parameters. Through the participants' answers to the post-evaluation questionnaire, we determined that CAT tools can have both a positive and negative impact on the achievement of accessibility, more specifically when users are not accessibility-savvy (Group 2). Both SDL Trados Studio 2017 and MemoQ v8.7 provide default settings that support certain features, but not all of them. In consequence, we can confirm hypothesis H2, which states that the functional completeness of the tool used has an impact on the final degree of the accessibility achieved. Similarly, we can conclude that CAT tools can be a useful instrument to employ for the localisation of accessibility features, more specifically to localisers who are not familiar with accessibility, but there are still several issues that users need to check in the target file.

5.1.3 Functional appropriateness

With the term *functional appropriateness*, we define the degree to which CAT tools facilitate the accomplishment of the selected tasks. In this sub-section, we describe the qualitative data collected through the answers related to the additional information offered by the CAT tools. Participants were asked whether the tool gave useful information about the features' context, whether any error messages were displayed, and whether they believed the tools could have a positive or negative impact when localising accessibility features (see questions A12-15 and B12-15 in Appendix F).

5.1.3.1 Additional information offered by the tools

Concerning SDL Trados Studio 2017, all participants in Group 1 (N=5) mentioned that they could retrieve additional information about the context of the feature they were dealing with. Among the features offered by the tool, four people (P03, P06, P08, and P10) mentioned the protected source code included in the tags as one of the features that helped them understand the feature's context. P03 explained that through the elements visible in the tags it was possible to "better understand the content", while P06 added that "the tool can show the extended version of tags so that their purpose (and that of the related text) can be better understood".

Various other features were indicated by participants from Group 1. P08 mentioned the different text format of links, which helped them identify this specific element, while one participant (P10) also highlighted the relevance of the attribute tab included in the editor, as it provided further details about the segment. Moreover, P07 mentioned that SDL Trados Studio 2017 as a whole was complete and precise, but did not specify which additional information or feature they believed was useful.

On the contrary, only three participants in Group 2 (P01, P04, and P09) answered positively to the first question and mentioned to have retrieved additional information about the feature's context. Again, the source code protected by tags was indicated by two participants (P01 and P04); however, P04 stated that, although they believed it was useful to have tags included in the translatable segment, they did not know what they meant. This might be due to the fact that P04 reported limited experience when dealing with HTML files, as they used CAT tools exclusively for translation projects and only localised a simple website during the MA Localisation class.

Moreover, P03 also added links different text format as a feature they found helpful in identifying the element. On the contrary, one participant (P05) stated that they were not sure whether the tool offered useful information, while P02 indicated that SDL Trados Studio 2017 did not offer any additional information about the features' context.

In the case of MemoQ v8.7, all participants in Group 1 (N=5) stated to have retrieved additional information about the accessibility feature's context. Similarly to the first tool, the source code included in the tags was mentioned by all of them. However, the majority (N=4) highlighted the importance of the preview pane. P03 explained: "I think the best feature that helped me when translating the file was the fact of showing the document structure. This way I could understand what kind of content I was translating, for instance, the alternative text for images".

Furthermore, almost all participants in Group 2 (N=4) indicated that they could identify the elements easily thanks to both the tags and the preview tab. Only one participant (P02) answered negatively to the question. Concerning the first feature, P04 also stated that it was easier to manage them compared to SDL Trados Studio 2017; while P05 mentioned that the preview mode helped them have an idea about the final product, and therefore it was easier to identify the accessibility features.

In general, we can state that more than half of the participants (N=6 for the first tool and N=7 for the second) suggested that both CAT tools offer features that can help identify the elements they need to localise and determine their context. The most important aspect highlighted by almost all participants was the visual component, as they could identify easily certain elements such as links or alternative texts. Again, we notice a difference between the two groups: participants in Group 2 (P02, P04, P05, and P09) either expressed their uncertainty in identifying the element's context, or did not retrieve any useful information, while Group 1 suggested that the tool help retrieve the important attributes. This might be due to the fact that participants from Group 2 were not aware of certain accessibility features and, therefore, did not pay attention to the additional information offered by the two tools related to these elements.

5.1.3.2 Error messages displayed by the tools

In the case of SDL Trados Studio 2017, almost all participants in Group 1 (N=4) stated to have got error messages. However, they stated that none of them were related to accessibility issues, as they concerned missing spaces, date formats, or various translation problems. Concerning MemoQ v8.7, two participants (P06 and P10) mentioned that the tool displayed error messages related to missing tags and links, respectively, while P08 mentioned that it displayed an error message related to space issues.

On the contrary, only one participant in Group 2 (P09) stated that both tools showed error messages, but they were not sure whether they related to accessibility. This might be due to the fact that P09 reported insufficient knowledge of accessibility and, consequently, does not have any experience with accessibility issues. Moreover, they also reported a limited experience in dealing with HTML files, as they specified in the preliminary questionnaire that they were not familiar with this type of document. The majority of participants (N=6), however, stated that the tool did not display any error message during the localisation process.

5.1.3.3 Tools' impact on the localisation of accessibility features.

The last question of the post-evaluation questionnaire was formulated as follows: 'Do you think that using this CAT tool could have a positive or negative impact when localising accessibility features? Please specify' (see questions A15 and B15 in Appendix F). Concerning SDL Trados Studio 2017, we received mixed responses from participants from Group 1: three participants (P03, P08, and P10) considered that the tool could have a positive impact on the localisation of accessibility features, while two (P06 and P07) stated the opposite. P03 indicated how the possibility "to modify the settings could improve the impact on localising accessibility features". However, we deem it relevant to highlight the fact that the three participants who gave a positive answer, also mentioned that the tool does not offer the possibility to transfer all the required elements. This aspect was the main reason that led P06 to give a negative answer: "Since not all localizable features show up in the editor, the use of this tool could lead to an incomplete localization of the accessibility features, resulting in a target less accessible than the source". We deem it relevant to recall that P06 works as a language engineer and may have a more professional outlook in comparison to students, who may not have experienced these issues in real-life situations. In addition, P07 compared the two tools and argued that SDL Trados Studio 2017 offered less useful features in comparison to MemoQ v8.7 and, therefore, it was not easy to produce an accessible target file.

The answers of Group 2 offered a different scenario: four participants (P01, P02, P04, and P05) considered that the tool could have a positive impact on the localisation of accessibility features, while only one person (P09) was not sure. P05 stated that it was easier to translate these elements thanks to the tool, and P02 highlighted the fact that the tool could help them save time. Another positive aspect was highlighted by P04, who mentioned that users are aware of the various features and, in consequence, it can be easier to identify them and localise them accordingly. P01 provided an example, as they indicated that the tool could be useful for the localisation of `alt` text for non-text content.

In the case of MemoQ v8.7, only one participant of Group 1 (P06) considered that the tool could have a negative impact on the achievement of accessibility and mentioned the same reason indicated for the first tool. The remaining participants (N=4) answered positively but also highlighted several negative factors. For instance, P08 argued that users must be aware of all the elements that they need to adapt before starting the localisation of the HTML file, while P10 considered SDL Trados Studio 2017 better compared to the second tool, as it provided more functionalities, such as the possibility to add and edit attributes in the Parser tab.

Moreover, P03 stated: "Even if I had to do manual changes to the target file using Notepad++, I consider that the tool was useful to translate many of the accessibility features, therefore I think it can have a positive impact in this kind of tasks". Last, P07 was the only participant in Group 1 that did not mention any negative aspect and underlined the importance of the preview pane.

In line with the results of the first tool, four participants of Group 2 (P01, P02, P04, and P05) considered that the tool could have a positive impact on the localisation of accessibility features, while only one person (P09) stated that the tool did not contribute in helping them localising accessibility feature. This might be because P09 did not have knowledge of accessibility and, while localising the HTML file, the tool did not help them identify the required elements that were listed in the post-evaluation questionnaire. Therefore, during the localisation process, the participant did not know whether the element they were localising was, in fact, an accessibility feature and, consequently, could not adapt them accordingly. On the other hand, among the positive aspect, P01 mentioned that MemoQ v8.7 was useful for the localisation of text alternatives, as the tool displayed them in the editor as normal segments. Moreover, P05 considered the tool more intuitive compared to SDL Trados Studio 2017, and, consequently, helped them identify the selected requirements easily.

To sum up, we can generalise by stating that Group 1, although the majority still argued that CAT tools can have a positive impact, stressed the shortcomings of both tools, while Group 2 judged that both systems could be a useful instrument to achieve accessibility. This difference can be explained by the fact that, since they are not accessibility-savvy, participants in Group 2 took into account that the tools can transfer automatically certain accessibility features they were not aware of. In contrast, participants in Group 1 focused on the final product and, since they detected several issues, they considered the fact that they still had to make further manual changes to obtain a fully accessible target file and, therefore, concluded that the tools may also have negative aspects.

5.1.3.4 Summary of the findings

Although participants pointed out several issues related to the missing support for certain features, the majority agreed on the positive impact that both CAT tools can have on the localisation of accessibility features. In the case of SDL Trados Studio 2017, seven participants answered positively, one was not sure, and two answered negatively; while in the case of MemoQ v8.7, eight participants answered positively, and two answered negatively.

The most useful additional information about the element's context was provided by the source code included in the tags. Participants also mentioned the importance of seeing the structure of the document in MemoQ's preview pane. This feature is also offered by SDL Trados Studio 2017, but it is not incorporated in the editor by default and, as stated by P04 during the experiment, the tab needs to be reloaded quite often to see the result. Moreover, MemoQ v8.7 was the only tool between the two systems to display any error message related to accessibility issues. For instance, P06 noticed that the tool warned them when a tag was missing, while P10 pointed out that the tool displayed an error message when the content of the link was incorrect (similar to the error message related to alternative texts illustrated in Chapter 4). As a result, we could state that participants believed that MemoQ v8.7 was of greater help for them in the identification and localisation of accessibility features when compared to SDL Trados Studio 2017 (although the quantitative results demonstrated the opposite). In conclusion, we can confirm H3, which states that the functional appropriateness of the tool used has an impact on the final degree of accessibility achieved, as both tools provided additional information to identify the feature's context and, as stated by participants, could help users localise accessibility features.

5.2 Influence of participants' knowledge of accessibility

The second independent variable that we took into consideration is the participants' degree of knowledge of accessibility. In Chapter 2, we reviewed a number of articles and studies that highlighted the role of localisers in the achievement of accessibility during the localisation process. We alluded to the fact that, to create a target file as accessible as the source, localisers should understand accessibility and know how to implement it through accessibility best practices. To analyse this aspect, we asked participants to examine the target files they produced using the two CAT tools and, if they considered it necessary, to make manual modifications to the code. We then observed the final target source codes (version 2) and compare them to the corresponding first target files (version 1) to determine how the participants' understanding of accessibility could improve its achievement. First, we took into consideration the selected requirement, to determine if participants were able to identify and modify them accordingly. Second, we analysed all the extra elements that were adapted into Italian and determined whether they concerned accessibility. If that was the case, we counted them as a 'manual modification' that improved the overall degree of accessibility conformance.

5.2.1 Analysis of participants' final target files

As we illustrated in Chapter 3, we divided the participants into two groups: one with basic knowledge of accessibility and experience with accessibility issues (Group 1), and one with no experience with accessibility issues and no or basic knowledge of accessibility (Group 2). In the following sub-sections, we present the findings collected through the analysis of the final target files for each group, respectively. Moreover, we illustrated the detailed results for every participant in Appendix G.

5.2.1.1 Results of Group 1

In the case of Group 1, as we pointed out in previous sections, two participants (P03 and P10) modified certain parameters in the settings of SDL Trados Studio 2017 and were able to transfer more elements in comparison to the rest of participants. We deem it fundamental to highlight this aspect, as it proves that being aware of the specific accessibility features, detected in the code during the initial observation, led to improving the degree of accessibility conformance from the beginning. In Table 5.7, we present the results of the analysis of the final target files (versions 2): in red we indicated both the manual modifications that the participants made in the code (related to the selected accessibility features) and the additional modifications.

SC	N° of total occurrences	GROUP 1					MEAN	SD
		P03	P06	P07	P08	P10		
1.1.1 Non-text Content	18	18	18	18	18	18	18	0
2.4.1 Bypass Blocks	3	3	3	3	3	3	3	0
2.4.2 Page Titled	1	1	1	1	1	1	1	0
2.4.4 Link Purpose (in Context)	46	46	46	46	46	46	46	0
3.1.1 Language of Page	1	1	1	1	1	1	1	0
3.1.2 Language of Parts	2	2	2	2	0	2	1.6	0.894
3.2.2 On Input	3	3	3	3	3	3	3	0
3.3.1 Error Identification	5	5	5	5	5	5	5	0
3.3.2 Labels or Instructions	1	1	1	1	1	1	1	0
Additional manual modifications	?	29	3	0	0	33	13	16.54
TOTAL	**80**	**109**	**83**	**80**	**78**	**113**	92.6	16.95

Table 5.7: Results of the analysis of the final target file (SDL Trados Studio 2017).

Almost all participants in Group 1 (N=4) modified all the selected requirements. Only one participant (P08) did not modify correctly the SC *Language of Parts*, although the participants stated to have changed it manually. This might be due to a mistake or inadvertence from the

participant's part but proves that they did identify the issue. Therefore, taking this aspect into account, we can state that all participants in Group 1 (N=5) were able to produce a fully accessible product. More specifically, they successfully completed the tasks they could not accomplish using the tool: the value of the `lang` attribute, indicating the language of the whole document, was set to Italian; while the values of the `lang` attributes, for the parts that were originally in French, were set to the corresponding locale the participants decided to translate into; finally, the `aria-label` attribute was also translated into Italian.

Furthermore, we considered the manual modifications (presented in the 11th row of Table 5.7) that three participants (P03, P06, and P10) made in addition to the changes related to the selected SC. As we explained in Section 5.2, we analysed all the extra elements that the participants localised and, if they concerned accessibility, we counted them as a 'manual modification'. These elements were not supported and automatically transferred by the tool, yet they were considered by participants as elements that could be adapted to the target locale to improve accessibility. This aspect will be discussed in more detail in Section 5.2.1.3, where we will give an overview of the additional features the participants decided to localise. To sum up, we can conclude that, through manual modifications, participants' success rate scores were around 100% (99.6%, SD=0.8), but through the additional manual modifications that we decided to include, some of the participants' success rate was over 100%.

Similarly, in the case of MemoQ v8.7, participants identified the elements that the tool did not automatically localise, and modified it accordingly. Again, as we can see in Table 5.8, only participant P08 could not successfully complete all the tasks, while the rest of participants' success rate score was 100%, with an average degree of accessibility conformance of 99.6% (SD=0.8). Because MemoQ v8.7 supports fewer elements compared to the first CAT tool, participants had to manually modify more features, such as SC *Language of Page* and *Labels or Instructions*. Concerning the additional elements that participants decided to localise, we notice that the same type of features was also modified for MemoQ v8.7 (see Section 5.2.1.3). Only in one case, the participant (P06) decided to change one element that they did not modify in the final target file for SDL Trados Studio 2017, which, in consequence, led to a higher mean success rate score compared to the mean score of SDL Trados Studio 2017. This might be because the participant realised later that an additional element could be modified, but decided to not change it in the other file due to time constraints.

SC	N° of total occurrences	GROUP 1					MEAN	SD
		P03	P06	P07	P08	P10		
1.1.1 Non-text Content	18	18	18	18	18	18	18	0
2.4.1 Bypass Blocks	3	3	3	3	3	3	3	0
2.4.2 Page Titled	1	1	1	1	1	1	1	0
2.4.4 Link Purpose (in Context)	46	46	46	46	46	46	46	0
3.1.1 Language of Page	1	1	1	1	1	1	1	0
3.1.2 Language of Parts	2	2	2	2	0	2	1.6	0.894
3.2.2 On Input	3	3	3	3	3	3	3	0
3.3.1 Error Identification	5	5	5	5	5	5	5	0
3.3.2 Labels or Instructions	1	1	1	1	1	1	1	0
Additional manual modifications	?	29	4	0	0	33	13.2	16.39
TOTAL	**80**	109	84	80	78	113	92.8	16.81

Table 5.8: Results of the analysis of the final target file (MemoQ).

5.2.1.2 Results of Group 2

Contrary to Group 1, all participants in Group 2 (N=5) decided to work with default settings for both SDL Trados Studio 2017 and MemoQ v8.7. In addition to the reason mentioned in Section 5.1.1.2, this aspect could be explained by either the fact that participants concluded that the default settings were sufficient to transfer the accessibility features they identified during the observation of the code, or the fact that they were not aware of certain elements that needed to be localised. The latter is also supported by the answers to the post-evaluation questionnaire (see Table 5.3 and Table 5.4), in which the majority of participants in Group 2 (N=4) was not sure whether they were able to localise certain elements, as they did not know them.

Furthermore, through the analysis of the final target files (version 2), we notice several issues that participants did not resolve. As we mentioned in previous sections, for both tools, all participants (N=10) stated that they could produce an accessible product. Moreover, all participants in Group 1 (N=5) indicated that they made further modifications in the exported file (which we confirmed in the previous section), while only one participant from Group 2 (P04) stated to have done so. In Table 5.9, we summarised the results we obtained from the analysis of the final target files (version 2) of Group 2 for both SDL Trados Studio 2017 and MemoQ v8.7 (as the results for each participant were the same for both tools). The data collected through this last analysis shows that the features that were not automatically transferred by the tool were not modified by the participants from Group 2 either.

87

SC	N° of total occurrences	GROUP 2					MEAN	SD
		P01	P02	P04	P05	P09		
1.1.1 Non-text Content	18	18	18	18	18	18	18	0
2.4.1 Bypass Blocks	3	3	3	3	3	3	3	0
2.4.2 Page Titled	1	1	1	1	1	1	1	0
2.4.4 Link Purpose (in Context)	46	46	46	46	46	46	46	0
3.1.1 Language of Page	1	0	0	0	0	0	0	0
3.1.2 Language of Parts	2	0	0	0	0	0	0	0
3.2.2 On Input	3	3	3	3	3	3	3	0
3.3.1 Error Identification	5	5	5	5	5	5	5	0
3.3.2 Labels or Instructions	1	0	0	1	0	0	0.2	0.447
Additional manual modifications	?	0	0	0	0	0	0	0
TOTAL	**80**	76	76	77	76	76	76.2	0.447

Table 5.9: Results of the analysis of the final target file (SDL Trados Studio 2017 and MemoQ).

When we examined the answers to the post-evaluation questionnaire related to the way participants localised the selected SC, only one participant (P04) indicated that they have made a manual change to the final target code (version 2). In fact, the only manual modification we identified in the final target files (version 2) for both tools was the `aria-label` attribute localised by said participant. This might be due to the fact that P04 reported basic knowledge of accessibility, as they attended the MA Localisation course, or to the fact that they detected the sentence and believed it would have been better to localise it into the target locale. On the contrary, although the rest of the participants (N=4) affirmed to have produced an accessible webpage, they did not modify any feature that instead should have been adapted to the target locale. Moreover, participants indicated that they were able to modify certain elements, such as SC *Language of Parts*, yet they were still in English in the final target file (version 2). In other cases, they realised afterwards, while they were completing the post-evaluation questionnaire, of certain elements they should have localised and, consequently, answered negatively to the question. This might be due to the lack of knowledge of accessibility: since the participants were not aware of several aspects that should have been taken into consideration, they were not able to change them and, subsequently, they produced a target file that was not fully accessible. In consequence, the degree of accessibility conformance achieved by the participants from Group 2 stayed the same as the one achieved using the two CAT tools: 95.2% (SD=0.4).

5.2.1.3 Additional findings

In Section 5.2.1.1, we identified a number of additional elements, other than the selected SC, which were localised by three participants of Group 1 (P03, P06, and P10) and that are related to accessibility: `input`'s `value` attribute, and `content` elements.

Concerning the first element, the `value` attribute of the `input` element, two participants (P03 and P10) opted to localise it into Italian. The value attribute specifies the content embedded in the input element, which, in turn, specifies the input field in which users can enter data. As we can see in Figure 5.2, in the source code the attribute repeats the sentence included in the text unit. Both tools do not support this feature by default, and it was not transferred in the target file. However, we again examined the Parser tab in SDL Trados Studio 2017 and concluded that users can add the `value` attribute to the 'input' rule and set it as translatable. In this specific case, these elements were modified manually by the participants.

Figure 5.2: Example of `value` attribute in an `input` element.

Furthermore, as we already mentioned in Section 4.1.2.3, there are localisable attributes that are embedded in the `meta` elements. However, both tools only displayed the `title` attribute and not the `content`. Again, we already illustrated that it is possible to add this attribute to the 'meta' rule in SDL Trados Studio 2017, but there are no settings in MemoQ v8.7 related to this type of element. In our case, participants P06 and P10 decided to localise the information included in the `content` attributes, as we can see in Figure 5.3.

Figure 5.3: Example of `content` attribute in a `meta` element.

5.2.2 Summary of the findings

Contrary to the results of the analysis of the target files exported from the two CAT tools (target version 1), we noticed a considerable difference between the final target files produced by Group 1 and Group 2 (target version 2). The latter, although the participants believed they produced and accessible product, did not make any manual modification to the final source code. The only exception being the localisation of the `aria-label` attribute made by participant P04. As a result, the success rate score for Group 2 did not change (60% SD=0, for both tools) in comparison to the success rate scored using the two CAT tools. On the contrary, nearly all participants of Group 1 (N=4) modified all the selected requirements and accomplished all the nine tasks. Only one participant (P08) did not localise the SC *Language of Parts* but was able to modify the other two elements (*Language of Page* and *Labels or Instructions*). In Figure 5.4, we indicated the percentage of accessibility achieved by every participant for both versions of the target file, taking into account also the additional manual modifications: the target file exported directly from the tool (lighter colour) and the final target file after manual changes (darker colour).
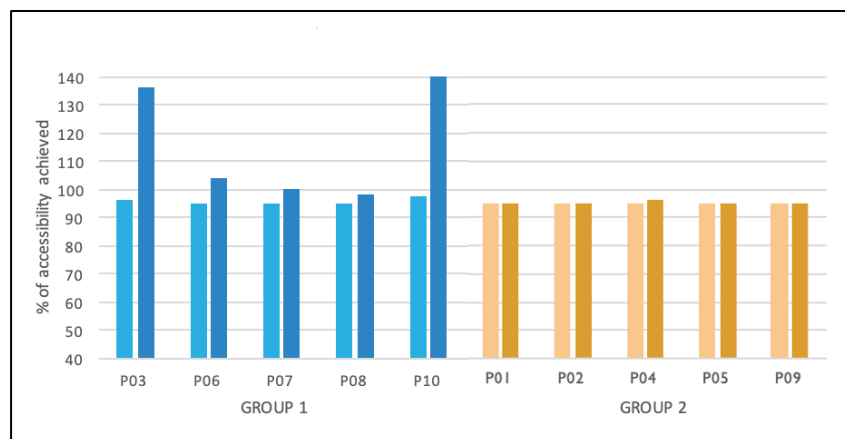
Figure 5.4: Percentage of accessibility achieved (SDL Trados Studio 2017 and MemoQ v8.7).

Therefore, we believe we can confirm hypothesis H4, which stated that the participants' level of knowledge of accessibility has an impact on the final degree of accessibility achieved. We observed that when localisers are accessibility-savvy and have already dealt with accessibility issues, they are more aware of (1) the elements they need to localise, and (2) any eventual issue they can encounter during the localisation process. A few participants in Group 1 (N=2) had already obtained a higher success rate score (70% and 80%, respectively) during the localisation of the HTML file using SDL Trados Studio 2017. In addition, three participants of the same group (P03, P06, and P10) modified several additional elements that are linked to accessibility,

proving the importance of understanding accessibility and knowing how to implement accessibility best practices.

5.3 User evaluation: summary of the main findings

In this last section, we briefly summarise the findings of the second stage of our research, namely the user evaluation.

In Section 5.1, we measured the three dependent variables to examine the effect caused by the first independent variable: CAT tools. To assess both SDL Trados Studio 2017 and MemoQ v8.7's *functional correctness*, we observed the target files produced by all the participants (N=10) using the tools (version 1). We calculated the percentage of accessibility achieved by every participant and determined the mean score. For the first tool we reviewed, SDL Trados Studio 2017, the mean score is 95,4% (SD=0,82), while for the second tool the mean score is 95%. This small difference might be due to the fact that two participants were able to apply the right settings. Through the measurement of functional correctness, we answered negatively to the second research question RQ2, as none of the participants transferred all the required accessibility features.

We also measured the tools' *functional completeness*, by calculating the success rate per participant. We divided the requirements into nine tasks and determined the mean success rate score for both tools: 63% and 60%, respectively. As we previously explained, the higher score of SDL Trados Studio 2017 is caused by the higher success rate of two participants. Through the assessment of functional completeness, we could support hypothesis H2 and determine that the feature and functionalities offered by the tools did not cover all the selected tasks. Therefore, the tools' functional completeness can have a negative impact on the final degree of accessibility achieved. However, we also determined that several participants, the majority being part of Group 2, were not aware of a number of accessibility features they had to localise during this experiment. Consequently, the tools' functional completeness can have a positive impact on localisers who are not accessibility-savvy, as the tools support and transfer the majority of selected accessibility features by default, which some of the participants were not aware of.

Lastly, we assessed the two CAT tools' *functional appropriateness*. Contrary to the two first variables, we adopted a qualitative approach to analyse the participants' answers to a post-evaluation questionnaire. Overall, the features that helped the participants identify and understand the specific elements were the source code included in the tags for SDL Trados

Studio 2017, and the preview pane for MemoQ v8.7. The majority of participants also stated CAT tools could have a positive impact during the localisation of accessibility features. However, several others, being part of Group 2, also answered negatively to the question, by highlighting the fact that the tools do not support all the features. Therefore, we could confirm H3, as the functional appropriateness of the tool does, in fact, influence the degree of accessibility achieved during the process.

In Section 5.2, we examined the second independent variable: the degree of knowledge of accessibility. We analysed the final target files produced by the participants. In conclusion, we stated that Group 1 could modify all the correct accessibility features that were not transferred by the tools. Moreover, three participants changed several additional elements (`value` and `content` attributes), which improved the overall percentage of accessibility achieved. In fact, the mean success rate score of Group 1 is 115.8% (SD=20,91) for SDL Trados Studio 2017 and 116% (SD=20.77) for MemoQ v8.7. On the other hand, Group 2 did not localise any of the selected requirements manually, except for one feature (`aria-label` attribute) modified by one participant. Therefore, the mean success rate score for both tools is 95.2% (SD=0,447). Through this analysis, we could support H4, and determine the relevance of understanding accessibility and accessibility best practices to obtain a target product as accessible as the source.

6 Conclusions

6.1 Overview of the research

In this Master's thesis, we examined the impact of CAT tools on the achievement of accessibility during the localisation process. More specifically, we analysed the performance of CAT tools in supporting and transferring accessibility information embedded in an HTML5 file.

Firstly, we provided an overview of the fundamental concepts of our research: localisation, accessibility, and CAT tools. We illustrated the major studies in these fields, that guided us in defining our hypothesis and designing our study, which consisted of two parts: a tool descriptive analysis and a user evaluation. Concerning the descriptive approach, the researcher examined two CAT tools taken into consideration and determined whether they supported and transferred correctly all the selected accessibility features. In the second stage of our research, a user evaluation was carried out with ten novice web localisers to determine both the functional suitability of the tools and the influence of the participants' degree of knowledge of accessibility on its achievement in the final target product.

6.2 Main findings

The current research has attempted to determine whether CAT tools have a positive or negative impact on the localisation of a selected number of accessibility features. By analysing this aspect from a theoretical and practical point of view, we obtained the following results.

First, concerning the descriptive approach, we determined that the two CAT tools selected for the study, SDL Trados Studio 2017 and MemoQ v8.7, are not able to support and transfer all the selected requirements, which confirmed our first hypothesis. Nevertheless, they did transfer the majority of selected SC. SDL Trados Studio 2017 could transfer eight of the nine SC, of which six could be transferred by default and two by manipulating the settings. The only issue concerned the SC *Language of Parts*. Similarly, MemoQ v8.7 could accomplish seven out of the nine tasks by using the default settings. In this case, issues include the lack of support of SC *Languages of Parts* and *Labels or Instructions*. Contrary to the first tool, MemoQ v8.7 is less flexible, as users are able to modify only the import settings; while in SDL Trados Studio 2017,

users can add and edit several attributes and elements that are embedded in the code. To sum up, the two CAT tools transferred more than 75% of selected requirements (around 77% per MemoQ v8.7 and 88% per SDL Trados Studio 2017), but not all of them. Therefore, we could not answer positively to the first research question.

Second, we examined the functional suitability of the two tools. Through the analysis of the target files produced by the ten participants who carried out the evaluation, we assessed the tools' functional correctness, functional completeness, and functional appropriateness. We determined that participants could not transfer all the requirements and complete all the tasks for two reasons: (1) the limited support offered by the tool chosen, which we have already pointed out during the first step; and (2) the limited knowledge and experience of the participants, which we further expanded on during the last part of our research. Although the results indicate that CAT cannot support some – although only a small number – features, we also highlighted some positive aspects. We determined that both tools offer useful information about the features' context in the form of tags and by showing the preview, two factors that helped participants identify the accessibility features easily. In addition, the majority of participants agreed on considering these tools as a support in localising certain elements. This second part of our research, therefore, provided both positive and negative outcomes. It confirmed the results of the descriptive approach, but also underlined some additional positive factors.

Finally, as the last step, we also analysed the influence of participants' knowledge of accessibility intending to determine whether it influenced the degree of accessibility achieved in the final target product. Participants were divided into two groups, one with basic experience on accessibility issues, and the other without. They were asked to hand in two versions of the target file: the one exported directly from the CAT tool (version 1) and a final target file where they could make manual changes if deemed appropriate (version 2). The first difference could be observed in the target files exported from SDL Trados Studio 2017, as two participants had manipulated the settings to improve the tool's performance in identifying and supporting the required features. But the most significant difference was noticed in the difference between the two groups' final product (version 2): Group 1 success score was much higher compared to Group 2, as one group scored over and the other less than 100%. This finding proves that when localisers are familiar with accessibility and know how to handle eventual problems related to accessibility, the overall degree of accessibility achieved can improve.

6.3 Limitations and future work

In this section, we aim at discussing possible limitations of this research and how they can be improved in future studies.

The test webpage, which was employed for both the first and second stage of our research, has certain limitations associated with it. First, although some of the selected SC comprised more than one technique, the code included only a limited number of sufficient techniques that could be tested. This aspect could be improved by creating a test webpage ad-hoc or a more comprehensive corpus, in which all the required techniques can be introduced to obtain a deep insight of the numerous accessibility-related best practices that can be implemented in a webpage. Furthermore, we based the selection of SC on previous studies that shared similarities with our research, which resulted in the selection of the nine requirements, despite the fact that numerous criteria included in the WCAG may relate to both localisation and accessibility, as shown in Section 2.3.3. The fact of making a subjective – although justified – choice and limit the selected range to only nine elements may be considered as a limitation in our research. Therefore, in future studies, a wider selection of criteria could be employed for this type of analysis.

Another acknowledged limitation concerns the object of our research. Our analysis was limited to only two tools, and, consequently, we could not give a generalised answer to the first research question. Due to the vast offer of CAT tools on the market, we opted to base our choice on a) the popularity of the tools and b) a previous Master's thesis (Castro Hernandez 2015), which laid the foundation for us to develop our own research and focus primarily on the issue we wanted to investigate. Moreover, the two tools employed have numerous similarities and produced similar results, which limited the discussion on possible different scenarios and outcomes. By using different tools, then, we may have obtained different results and a more comprehensive overview of the current industry.

Concerning the user evaluation, the inclusion of a greater number of participants would have been beneficial to the research, but time constraints and the small number of people interested in our study did not allow us to include more participants. Moreover, there are numerous variables that we need to acknowledge that influenced the results. First, the participants' experience in the localisation industry. As illustrated in Chapter 3, nearly all participants were currently students at the Faculty of Translation and Interpreting and, consequently, did not have any professional experience in the localisation field. Second, the participants' familiarity with

the two CAT tools posed several issues. As stated in Chapter 3, a few participants mentioned that they did not use either of the two tools on a daily basis and, when they did, they did not use them for localisation projects. Therefore, it would be interesting to carry out a similar evaluation with people in the industry. Another factor concerns the language pair, as all the participants were native Italian speakers. By including people with different language skills, we may have obtained different results.

6.4 Research contribution

Despite the limitations acknowledged in the previous section, this Master's thesis aimed to contribute to the yet scarce literature on web localisation and web accessibility, focusing more specifically on the relation between accessibility standards and multilingual websites, and on the process of adaptation of certain accessibility features. As illustrated in Chapter 2, in recent years the researchers' interest shifted towards this specific interaction, which, as a result, presented the evidence of numerous issues encountered in both the public and private sector. With this research, we wanted to contribute to the field by focusing on the localisation process rather than on the final localised product, to highlight any possible issue that can undermine the achievement of accessibility. The findings, which proved that CAT tools do not support and transfer certain accessibility features, also support the argument which considers multilingual accessibility as a problem of increasing relevance.

Furthermore, the researcher aimed to contribute to the discussion on the role of localisers in the achievement of accessibility during the localisation process. Localisers play a central role in the adaptation of content from a locale to another and, therefore, they can be the bridge between the accessibility features embedded in the source code and the corresponding accessibility features that will be localised in the target file. Through the results related to the participants' degree of knowledge of accessibility, we proved that localisers who are accessibility-savvy can produce and improve the overall quality of HTML files in terms of accessibility.

References

Alcina, A. (2008). Translation technologies Scope, tools and resources. *Target*, *20*, 79–102. https://doi.org/10.1075/target.20.1.05alc

Altman, B. M. (2001). Definitions, Models, Classifications, Schemes, and Applications. In *Handbook of Disability Studies* (pp. 97–122). https://doi.org/10.4135/9781412976251

Bordandini, S. (2019). *Data Exchange Standards for Localization: Testing XLIFF Conformance in CAT Tools and TMS* (University of Geneva). Retrieved from https://archive-ouverte.unige.ch/unige:123917?all_subtypes=1

Bowker, L. (2002). *Computer-aided Translation Technology: A Practical Introduction*. University of Ottawa Press.

Casalegno, E. (2018). *Usability of Partially Localised Websites in Switzerland: A Study with Screen Reader Users* (University of Geneva). Retrieved from https://archive-ouverte.unige.ch/unige:111753

Castro Hernandez, L. (2015). *A comparative evaluation of CAT tools during the localisation of HTML5 files* (University of Geneva). Retrieved from https://archive-ouverte.unige.ch/unige:80665

Chisholm, W., Vanderheiden, G., & Jacobs, I. (1999). Web Content Accessibility Guidelines 1.0. Retrieved 19 November 2019, from https://www.w3.org/TR/WAI-WEBCONTENT/

Doherty, S. (2016). The impact of translation technologies on the process and product of translation. *International Journal of Communication*, *10*, 969.

Doyle, J. (2019). A Complete Overview of Canada's Accessibility Laws. Retrieved 19 November 2019, from Siteimprove website: https://siteimprove.com/en-ca/blog/a-complete-overview-of-canada-s-accessibility-laws/

Dulock, H. L. (1993). Research Design: Descriptive Research. *Journal of Pediatric Oncology Nursing*, *10*(4), 154–157. https://doi.org/10.1177/104345429301000406

Dunne, K. J. (2006). *Perspectives on localization*. Retrieved from http://data.rero.ch/01-R004170943/html?view=GE_V1

EAGLES (Ed.). (1996). *Evaluation of natural language processing systems: Final report* (Version of October 1996). Copenhagen: Center for Sprogteknologi.

EAGLES. (1999). The EAGLES 7-step recipe. Retrieved 4 November 2019, from https://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html

EAGLES. (2019). What is EAGLES? Retrieved 20 November 2019, from EAGLES website: http://www.ilc.cnr.it/EAGLES96/edintro/node9.html#SECTION00042100000000000000

Esselink, B. (2000). *A practical guide to localization* ([Revised ed.].). Retrieved from http://data.rero.ch/01-R003004624/html?view=GE_V1

Esselink, B. (2003). Localisation and translation. In H. Somers (Ed.), *Computers and*

Translation: A translator's guide (pp. 67–86). John Benjamins Publishing.

European Commission. (2019). *Digital inclusion and web accessibility in the European Union.* Retrieved from https://ec.europa.eu/digital-single-market/en/news/digital-inclusion-and-web-accessibility-brochure

Federico, M., Cattelan, A., & Trombetti, M. (2012). Measuring User Productivity in Machine Translation Enhanced Computer Assisted Translation. *Proceedings*, 10. Retrieved from http://www.mt-archive.info/AMTA-2012-Federico.pdf

Flórenz, S., & Alcina, A. (2015). A Quality Model for the Evaluation of Open Translation Technologies. In P. Sandrini & M. G. González (Eds.), *Translation and Openness* (1st ed., pp. 81–115). https://doi.org/10.15203/2936-88-2-5

GALA. (2015). What is Localization? Retrieved 8 June 2019, from GALA Global website: https://www.gala-global.org/industry/intro-language-industry/what-localization

GALA. (2015a). What is Globalization? Retrieved 10 June 2019, from GALA Global website: https://www.gala-global.org/language-industry/intro-language-industry/what-globalization

GALA. (2015b). What is Internationalization? Retrieved 10 June 2019, from GALA Global website: https://www.gala-global.org/language-industry/intro-language-industry/what-internationalization

Garcia, I. (2014). Computer-Aided Translation. In S. Chan (Ed.), *The Routledge Encyclopedia of Translation Technology*. https://doi.org/10.4324/9781315749129.ch3

Goggin, G. (2018). Disability and Digital Inequalities: Rethinking Digital Divides with Disability Theory. In M. Ragnedda & G. W. Muschert, *Theorizing Digital Divides* (1 Edition). London : New York: Routledge, Taylor & Francis Group.

Government of Canada. *Standard on Web Accessibility*, (2011).

Government of Canada. *Directive on Official Languages for Communications and Services.* , (2012).

Gow, F. (2003). *Metrics for Evaluating Translation Memory Software*.

Gutiérrez y Restrepo, E., & Martínez Normand, L. A. (2010). Localization and web accessibility. *Tradumàtica: Tecnologies de La Traducció*, *8*(1). https://doi.org/10.5565/rev/tradumatica.106

Henry, S. L., Abou-Zahra, S., & Brewer, J. (2014). The Role of Accessibility in a Universal Web. *Proceedings of the 11th Web for All Conference*, 17:1–17:4. https://doi.org/10.1145/2596695.2596719

Höge, M. (2002). *Towards a framework for the evaluation of translators' aids systems* (University of Helsinki). Retrieved from http://urn.fi/URN:ISBN:952-10-0556-4

Hutchins, J. (1998). The Origins of the Translator's Workstation. *Machine Translation*, *13*(4), 287–307.

International Telecommunication Union (ITU). (2018). *Measuring the Information Society Report* (Vol. 1). Geneva.

ISO. (1999). ISO/IEC 14598-1:1999. Retrieved 25 November 2019, from ISO website: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/02/49/24902.html

ISO. (2001). ISO/IEC 9126-1:2001. Retrieved 25 November 2019, from ISO website: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/02/27/22749.html

ISO. (2015). ISO 17100:2015(en), Translation services—Requirements for translation services. Retrieved 10 November 2019, from Online Browsing Platform (OBP) website: https://www.iso.org/obp/ui/#iso:std:iso:17100:ed-1:v1:en

ISO 25000. (2019). ISO 25010. Retrieved 25 November 2019, from https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

ISO/IEC. (2011). ISO/IEC 25010:2011. Retrieved 20 November 2019, from ISO website: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/03/57/35733.html

ISO/IEC. (2011a). ISO/IEC 25040:2011. Retrieved 25 November 2019, from ISO website: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/03/57/35765.html

ISO/IEC. (2014). ISO/IEC 25000:2014. Retrieved 25 November 2019, from ISO website: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/47/64764.html

Jiménez-Crespo, M. A. (2013). *Translation and web localization*. London: Routledge.

Kenny, D. (1999). CAT Tools in an Academic Environment—What Are They Good For? *Target*, *11*(1), 65–82.

King, M. (2005). Accuracy and Suitability: New Challenges for Evaluation. *Language Resources and Evaluation*, *39*(1), 45–64. Retrieved from JSTOR.

Kirkpatrick, A., O Connor, J., Campbell, A., & Cooper, M. (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Retrieved 23 September 2019, from W3C Recommandation website: https://www.w3.org/TR/WCAG21/

Krüger, R. (2016). Contextualising Computer-Assisted Translation Tools and Modelling Their Usability. *Trans-Kom - Journal of Translation and Technical Communication Research*, *9*(1), 114–148.

Lawson, B., & Sharp, R. (2010). *Introducing HTML5*. Pearson Education.

Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research Methods in Human-Computer Interaction*. Saint Louis: Morgan Kaufmann.

Lubbers, P., Albers, B., & Salim, F. (2010). *Pro HTML5 Programming: Powerful APIs for Richer Internet Application Development*. Apress.

Melby, A. K. (2006). MT+TM+QA: The future is ours. *Tradumàtica: Traducció i Tecnologies de La Informació i La Comunicació*, *0*(4). Retrieved from https://www.raco.cat/index.php/Tradumatica/article/view/56004

MemoQ. (2018). Hello, memoQ 2015! Retrieved 24 November 2019, from Translation software—MemoQ website: https://www.memoq.com/memoq-versions/hello-memoq-2015

MemoQ. (2019). Translators' essential. Retrieved 24 November 2019, from Translation

software—MemoQ website: https://www.memoq.com/translators-essentials

Morado Vázquez, L. (2012). *An empirical study on the influence of translation suggestions' provenance metadata* (University of Limerick). Retrieved from https://www.academia.edu/3409996/An_empirical_study_on_the_influence_of_translation_s uggestions_provenance_metadata

Ó Broin, U. (2004). Accessibility is just another language: The common concerns of localization and accessibility. *Multilingual Computing and Technology*, *15*(3), 17–20.

O Connor, J. (2012). *Pro HTML5 accessibility: Building an inclusive web*. [Berkeley, CA] : New York: Apress ; Distributed to the book trade worldwide by Springer Science+Business Media.

OASIS. (2006). XLIFF 1.2 Representation Guide for HTML. Retrieved 19 November 2019, from http://docs.oasis-open.org/xliff/v1.2/xliff-profile-html/xliff-profile-html-1.2.html

OASIS. (2014). XLIFF Version 2.0. Retrieved 19 November 2019, from http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html

Oliver, M. (1996). Defining Impairment and Disability: Issues at stake. In C. Barnes & G. Mercer (Eds.), *Exploring the divide: Illness and disability* (pp. 29–54). Leeds: Disability Press ; University of Leeds.

Parliament of Canada. *Accessible Canada Act—An Act to ensure a barrier-free Canada*. , Pub. L. No. C–81 (2019).

Petrie, H., Savva, A., & Power, C. (2015). Towards a Unified Definition of Web Accessibility. *Proceedings of the 12th Web for All Conference*, 35:1–35:13. https://doi.org/10.1145/2745555.2746653

Pontus, V. (2019). *Evaluating the Accessibility of Localised Websites: The Case of the Airline Industry in Switzerland* (University of Geneva). Retrieved from https://archive-ouverte.unige.ch/unige:126286

Pym, A. (2004). *The moving text: Localization, translation and distribution*. Retrieved from http://data.rero.ch/01-R003649543/html?view=GE_V1

Pym, A. (2011). Website Localizations. In K. Malmkjær & K. Windle (Eds.), *The Oxford Handbook of Translation Studies*. https://doi.org/10.1093/oxfordhb/9780199239306.013.0028

Qadri, J. A., & Banday, M. T. (2009). Web Accessibility—A timely recognized challenge. *The Business Review*, *14(1 & 2)*, 99–102.

Quah, C. K. (2006). *Translation and Technology*. https://doi.org/10.1057/9780230287105

Raggett, D., Lam, J., Kmiec, M., & Alexander, I. (1998). *Raggett on HTML4* (Addison Wesley Longman). Retrieved from https://www.w3.org/People/Raggett/book4/ch02.html

Raya, R. M. (2004). Maxprograms—Use XLIFF to Translate Documents. Retrieved 22 November 2019, from Maxprograms website: https://www.maxprograms.com/articles/xliff.html

Rico, C. (2000). Evaluation Metrics for Translation Memories. *Language International*, *12*(6),

36–37.

Rico, C. (2001). *Reproducible models for CAT tools evaluation: A user-oriented perspective.* 12.

Rodríguez, M., Oviedo, J. R., & Piattini, M. (2016). Evaluation of Software Product Functional Suitability: A Case Study. *Software Quality Professional Magazine*, *18*(3). Retrieved from https://secure.asq.org/perl/msg.pl?prvurl=http://asq.org/software-quality/2016/06/software-quality/software-quality-management-evaluation-of-software-product-functional-suitability-a-case-study.pdf

Rodríguez Vázquez, S. (2013). Towards Defining the Role of Localisation Professionals in the Achievement of Multilingual Web Accessibility. *Tradumàtica: Tecnologies de La Traducció*, (11), 383–388. https://doi.org/10.5565/rev/tradumatica.37

Rodríguez Vázquez, S. (2016). *Assuring accessibility during web localisation: An empirical investigation on the achievement of appropriate text alternatives for images* (University of Geneva). Retrieved from https://archive-ouverte.unige.ch/unige:84104

Rodríguez Vázquez, S., & O'Brien, S. (2017). Bringing Accessibility into the Multilingual Web Production Chain: Perceptions from the Localization Industry. In M. Antona & C. Stephanidis (Eds.), *Universal Access in Human–Computer Interaction. Design and Development Approaches and Methods* (pp. 238–257). https://doi.org/10.1007/978-3-319-58706-6_20

Rodríguez Vázquez, S., & Torres del Rey, J. (2014). Fostering accessibility through web localization. *MultiLingual Magazine*, *25*(7), 32–37.

Sandelowski, M. (2000). Combining Qualitative and Quantitative Sampling, Data Collection, and Analysis Techniques in Mixed-Method Studies. *Research in Nursing & Health*, *23*(3), 246–255. https://doi.org/10.1002/1098-240X(200006)23:3<246::AID-NUR9>3.0.CO;2-H

Sandrini, P. (2008). Localization and Translation. *MuTra Journal*, *2*, 1–25.

SDL Trados. (2017a). SDL Trados Studio 2017 SR1. Retrieved 24 November 2019, from https://sdl.ssl.cdn.sdlmedia.com/636437364383099297EQ.pdf

SDL Trados. (2017b). SDL Trados Studio 2017. Retrieved 24 November 2019, from SDL* Documentation Center website: https://docs.sdl.com/LiveContent/web/pub.xql?action=home&pub=SDL%20Trados%20Studi o%20Help-v4&lang=en-US#docid=GUID-3618997B-F745-43DF-A22F-74C348373938&addHistory=true&query=&scope=&tid=&filename=GUID-2B372DA3-9DF8-47FA-AD36-8DB5668AEBA4.xml&resource=&inner_id=&toc=false&eventType=lcContent.loadDocGUI D-3618997B-F745-43DF-A22F-74C348373938

Shakespeare, T. (2010). The Social Model of Disability. In L. J. Davis (Ed.), *The Disability Studies Reader* (pp. 266–273). New York: Routledge, Taylor & Francis Group.

Somers, H. (2003). Translation memory systems. In H. Somers (Ed.), *Computers and Translation: A translator's guide* (pp. 31–48). Retrieved from https://benjamins.com/catalog/btl.35

Starlander, M. (2015). *Let the EAGLES Fly into New Standards: Adapting our CAT Tool Evaluation Methodology to the ISO 25000 Series*. Retrieved from https://archive-ouverte.unige.ch/unige:78023

Starlander, M., & Morado Vázquez, L. (2013). Training translation students to evaluate CAT tools using Eagles: A case study. In *Aslib: Translating and the Computer 35*. Retrieved from https://archive-ouverte.unige.ch/unige:35622

Torres del Rey, J., & Morado Vázquez, L. (2019). Transferring web accessibility through localization and internationalization standards. *The Journal of Internationalization and Localization*, *6*(1), 1–24. https://doi.org/10.1075/jial.19002.tor

Torres del Rey, J., & Rodríguez Vázquez, S. (2016). New insights into translation-oriented, technology-intensive localiser education: Accessibility as an opportunity. In *Proceedings of the 4th International Conference Technological Ecosystems for Enhancing Multiculturality (TEEM '16)*. https://doi.org/10.1145/3012430.3012634

Torres Hostench, O., Gil, J., Cid-Leal, P., Mor, A., Lao, B., Orozco-Jutorán, M., & Gijón, P. (2010). *TRACE: Measuring the impact of CAT tools on translated texts*.

United Nations. (2006). *Convention on the Rights of Persons with Disabilities*. 28.

United Nations, & Department of Economic and Social Affairs. (2019). *Disability and development report: Realizing the sustainable development goals by, for and with persons with disabilities : 2018*.

Van Dijk, J. (2005). *The deepening divide: Inequality in the information society*. Thousand Oaks, Calif: Sage Pub.

Vanderheiden, G. (1998). Universal Design and Assistive Technology in Communication and Information Technologies: Alternatives or Complements? *Assistive Technology : The Official Journal of RESNA*, *10*, 29–36. https://doi.org/10.1080/10400435.1998.10131958

W3C. (2007). HTML Design Principles. Retrieved 11 November 2019, from https://www.w3.org/TR/html-design-principles/#universal-access

W3C. (2018). HTML 5.3: 1. Introduction. Retrieved 23 June 2019, from https://www.w3.org/TR/html53/introduction.html#introduction

W3C. (2018a). HTML 5.3: 3. Semantics, structure, and APIs of HTML documents. Retrieved 23 June 2019, from https://www.w3.org/TR/html53/dom.html#elements-semantics

W3C/WAI. (2017). Diverse Abilities and Barriers. Retrieved 4 November 2019, from Web Accessibility Initiative (WAI) website: https://www.w3.org/WAI/people-use-web/abilities-barriers/

W3C/WAI. (2017a). Tools and Techniques. Retrieved 3 November 2019, from Web Accessibility Initiative (WAI) website: https://www.w3.org/WAI/people-use-web/tools-techniques/

W3C/WAI. (2017b). Accessible Rich Internet Applications (WAI-ARIA) 1.1. Retrieved 23 June 2019, from https://www.w3.org/TR/wai-aria-1.1/

W3C/WAI. (2018a). Essential Components of Web Accessibility. Retrieved 11 June 2019, from Web Accessibility Initiative (WAI) website: https://www.w3.org/WAI/fundamentals/components/

W3C/WAI. (2019). Introduction to Web Accessibility. Retrieved 11 June 2019, from Web Accessibility Initiative (WAI) website: https://www.w3.org/WAI/fundamentals/accessibility-intro/

W3C/WAI. (2019a). Participating in WAI. Retrieved 11 June 2019, from Web Accessibility Initiative (WAI) website: https://www.w3.org/WAI/about/participating/

W3C/WAI. (2019b). Understanding Techniques for WCAG Success Criteria. Retrieved 23 June 2019, from https://www.w3.org/WAI/WCAG21/Understanding/understanding-techniques

W3C/WAI. (2019d). Understanding Success Criterion 2.4.1: Bypass Blocks. Retrieved 22 November 2019, from https://www.w3.org/WAI/WCAG21/Understanding/bypass-blocks.html

W3C/WAI. (2019e). Understanding Success Criterion 3.1.1: Language of Page. Retrieved 22 November 2019, from https://www.w3.org/WAI/WCAG21/Understanding/language-of-page.html

W3C/WAI. (2019f). Understanding Success Criterion 3.2.2: On Input. Retrieved 22 November 2019, from https://www.w3.org/WAI/WCAG21/Understanding/on-input.html

W3C/WAI. (2019g). Understanding Success Criterion 3.3.2: Labels or Instructions. Retrieved 22 November 2019, from https://www.w3.org/WAI/WCAG21/Understanding/labels-or-instructions

Williams, C. (2007). Research Methods. *Journal of Business & Economics Research (JBER)*, *5*(3). https://doi.org/10.19030/jber.v5i3.2532

World Health Organization (WHO) (Ed.). (2011). *World report on disability*. Geneva, Switzerland: World Health Organization.

Yao, S. (2017). Application of Computer-aided Translation in English Teaching. *International Journal of Emerging Technologies in Learning (IJET)*, *12*(08), 105–117.

Appendices

Appendix A.

New elements introduced in the HTML5 specification:

Element	Description
`<section>`	generic document or application section
`<article>`	an independent piece of content of a document
`<aside>`	a piece of content slightly related to the rest of the document
`<hgroup>`	header of a section
`<header>`	group of introductory or navigational aids
`<footer>`	can contains information about the author, copyright, etc.
`<nav>`	section intended for navigation
`<figure>`	piece of self-contained flow content
`<figcaption>`	caption for an image or other graphical content
`<video>` and `<audio>`	used for multimedia content
`<track>`	text tracks for the video element
`<embed>`	plugin content
`<mark>`	text highlighted or marked for reference purposes
`<progress>`	completion of a task
`<meter>`	measurement
`<time>`	date and/or time
`<ruby>`, `<rt>` and `<rp>`	mark up ruby annotations
`<bdi>`	span text isolated from its surroundings for the purpose of bidirectional text formatting
`<wbr>`	line-break opportunity
`<canvas>`	dynamic bitmap graphics on the fly
`<command>`	command the user can invoke
`<details>`	additional information or commands
`<keygen>`	control for generating a *key pair*
`<output>`	type of output (e.g. calculation through scripting)
`<input>`	used for inputting data
global attributes	`accesskey, class, dir, id, lang, style, tabindex, title, contenteditable, contextmenu, data, draggable, hidden, role, aria-, spellcheck`

Adapted from "Pro HTML5 Accessibility", O'Connor, 2012

Appendix B.

Test webpage

The final test webpage's code can be viewed here:

https://drive.google.com/file/d/1U7Ovx-
ts5oQrC2tG57XRSb54v9Y7jisK/view?usp=sharing

Appendix C.

Background questionnaire
Thank you for your interest in the study. The following questionnaire consists of four sections and will take a maximum of 10 minutes to complete.
There are 26 questions in this survey.

Consent form
I have read and understood the information provided about the study. *

❏Yes

❏No

I confirm that I am over 18 years old *

❏Yes

❏No

I am aware that I will be asked to complete some tasks and express my opinions. *

❏Yes

❏No

I have understood that I can withdraw participation at any time. *

❏Yes

❏No

I am aware that my data will be treated confidentially. *

❏Yes

❏No

I consent to take part in this research study. *

❏Yes

❏No

Please, enter your email address:

_____

Background questions
What is your native language? (Check all that apply) *
Please choose all that apply:

❏Italian

❏English

❏French

❏Spanish

❏German

❏Other:

Other than your native language, please select which other languages you are fluent in (Check all that apply) *
Please choose all that apply:

❏Italian

❏English

❏French

❏German

❏Spanish

❏Other:

Are you currently studying at the Faculty of Translation and Interpretation? *

❏Yes

❏No

If yes, what is your current year of studies?
Please write your answer here:
If no, what is your current occupation?
Please write your answer here:

Research foundation
Are you familiar with localisation? *

❏Yes

❏No

If yes, have you ever localised a web file? Please specify.
Please write your answer here:
Are you familiar with web accessibility? *

❏Yes

❏No

If yes, have you ever dealt with accessibility issues while working on a localisation project? Please specify.

Please write your answer here:

Experience with CAT tools

Have you ever used a Computer-Aided Translation (CAT) tool? *

❏Yes

❏No

If yes, how often do you use it?

❏Rarely

❏Sometimes

❏Quite often

❏Often

❏For every translation project

Have you used SDL Trados Studio 2017 before? *

❏Yes

❏No

If yes, how familiar are you with this tool? (1 being 'not familiar' and 5 being 'expert user')

❏1 ❏2 ❏3 ❏4 ❏5

And if yes, what kind of project did you use this CAT tool for (technical translations, localisation, etc.)?
Please write your answer here:

Have you used MemoQ v8.7 before? *

❏Yes

❏No

If yes, how familiar are you with this tool? (1 being 'not familiar' and 5 being 'expert user')
Please choose only one of the following:

❏1❏2❏3❏4❏5

And if yes, what kind of project did you use this CAT tool for (technical translations, localisation, etc)?
Please write your answer here:
Have you ever heard of HyperText MarkUp Language (HTML)? *

❏Yes

❏No

If yes, have you ever worked with it (for example, created a webpage or modified the code with the assistance of a web design tool or source code editor such as Notepad++)? Please specify.
Please write your answer here:

Thank you for your time and for your collaboration.

Isotta Pacati

Appendix D.

CALL FOR PARTICIPATION

We are looking for participants who would be interested in taking part in a web localisation related research study, which will be part of a project for a master's thesis. In the section below you will find information about how the study works, the tasks you will be asked to conduct, and a link to a preliminary questionnaire.

The main purpose of this study is to determine if Computer-Aided Translation (CAT) tools can support and transfer the information concerning accessibility embedded in an HTML file, in the context of web localisation. With the term 'accessibility', we refer to the quality of a product, in our case a website, of being easily used by everyone, including people with disabilities.

You will be asked to localise a web page (HTML file) with two CAT tools from English to your native language. Finally, you will be asked to answer a questionnaire related to the task you conducted. The experiment will consist of one session of 1h30 maximum and it will take place in the computer rooms (Uni Mail, 6th floor, University of Geneva).

We will not test your abilities or knowledge of accessibility or localisation, nor will we take into account the quality of the final translation. The ultimate goal of the study is to determine the degree of accessibility support offered by the tools tested taking into account the target file that the participant will produce.

There are no potential risks involved in participating in this study. The participation in the study is voluntary, therefore you may withdraw at any time without penalty.

Your responses will be anonymous and all data collected will be treated confidentially. The lecturer of this course, Dr. Lucia Morado Vazquez, is not supervising this project, so your participation in the study will not have any impact (positive or negative) in your final grade.

If you are interested in this study, please fill in the preliminary questionnaire and once it is completed, send an email to isotta.pacati@etu.unige.ch.

https://formulaire.unige.ch/outils/limesurveyfac/traduction-interpretation/index.php/398172?lang=en

Thank you for your time.

Isotta Pacati

Appendix E.

CAT tool evaluation – Instructions

This study aims to determine if CAT tools can support and transfer the **accessibility information** embedded in an HTML code. You will be asked to localise an actual HTML code with two CAT tools, **SDL Trados Studio 2017** and **MemoQ v8.7**, and then answer a questionnaire related to the task you conducted.

1) Before starting with the actual localisation, <u>open the HTML file</u> (website_thesis_pacati.htm) with Notepad++ and <u>examine it</u>. In the code, you will find several 'accessibility features', namely the elements that help achieve accessibility that can be embedded in the code, such as coding elements, attributes, text units, etc.

2) <u>Start the CAT</u> tool and create a new project:

- Name the project **Experiment_CATtool_YourNumber** (e.g. Experiment_SDLTrados_01). [The number will be assigned to you by the research at the beginning of the evaluation.]
- Language Pair: select English (CA) as the source language and your native language as the target language.
- Add the source file: select **testwebsite_thesis_pacati.htm** and add it as the source file.
- Create a Translation Memory and a Termbase (these two elements will not be taken into account during the final evaluation).

⚠ Remember to change any eventual setting that, as far as you are concerned, is related to accessibility before finishing creating the project.

For **SDL Trados Studio 2017**:

When adding the source file, select 'File Types' > HTML5

Project Files
Add project files or folders and verify that all file types are correctly identified.

	Name	Size	Type	Usage
Experiment_SDLTrados_P10	corpus_thesis_pacati.htm	37 KB	HTML 5	Translatable
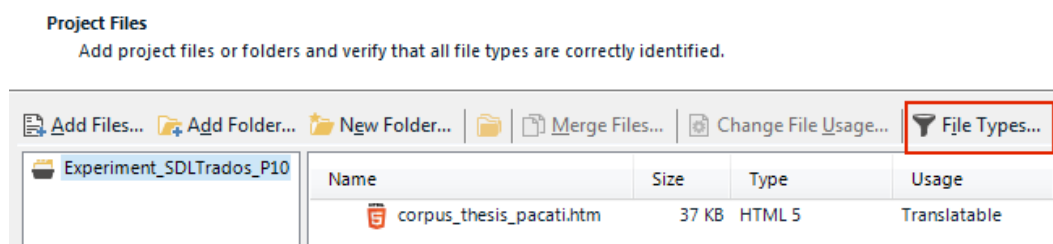
In the 'Project File Type Settings', you will find the following settings, where you can modify or add any eventual setting.

For **MemoQ v8.7**:

When adding the source file, select 'import with options' and verify that the filter is set as 'HTML filter'.
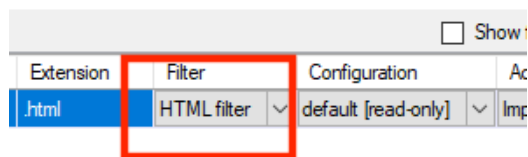
Click on 'Change filter and configuration'. The following tab will open and you will be able to modify any eventual setting.
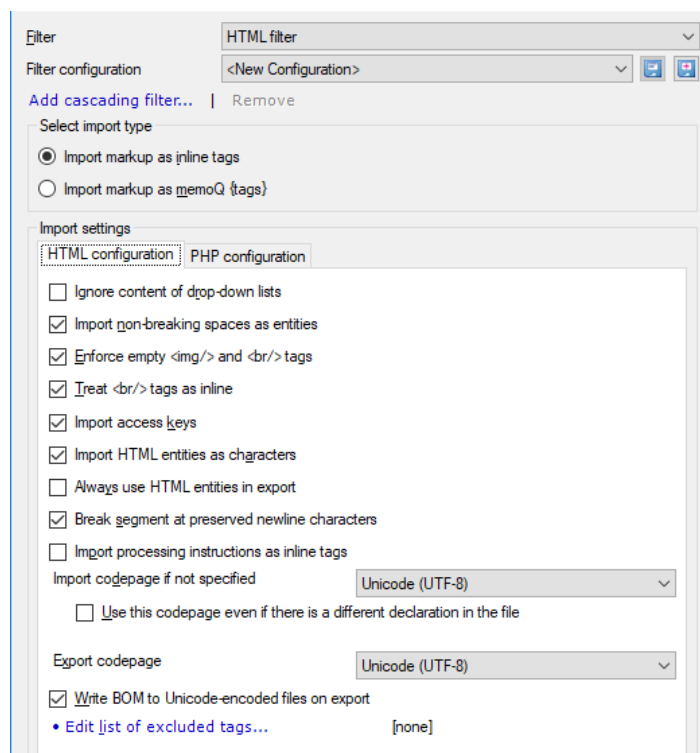
3) Localise the HTML code

- Make sure the tool shows the extended tags (set the tags length to 'long').
- Translate all the segments in the file website_thesis_pacati.

- You will find a few segments in French. Please translate them in another language other than your native language (for instance, in English).
- Make sure you confirmed all the segments and there are not any error messages.
- Once you are finished with the translation, finalise it if necessary and export the file in the source format (HTML) with the name **Target_CATtool_YourNumber.htm** (e.g. Target_SDLTrados_01.htm).

4) As a last step, you may underline examine the target HTML file you have just localised, to check if all the localisable elements have been translated correctly.

- Open the target file with Notepad++ and modify the code if you consider it necessary.
- Save the modified file with the name **Final_CATtool_YourNumber.htm** (e.g. Final_SDLTrados_01.htm).

[Points 1) to 4) will have to be repeated for the other CAT tool as well.]

5) Fill in the questionnaire post-evaluation.

Once you have completed the questionnaire and submitted it, send a zipped folder with the target and final files (with the name **Evaluation_YourNumber.zip**) to isotta.pacati@etu.unige.ch.

Thank you for your collaboration,

Isotta Pacati

Appendix F.

Questionnaire

post-evaluation

Thank you for taking part in this experiment. In this questionnaire, you will find two sections, respectively about SDL Trados Studio 2017 and MemoQ v8.7. Please answer the questions after completing the task with both tools and try to be as specific as possible.

If you need any help, please do not hesitate to ask the researcher.

There are 31 questions in this survey.

Please indicate your participant number:

SDL Trados Studio 2017

A1. Were you successful in producing an accessible target file? *

❏Yes

❏No

A2. Did you have to have to make any manual modification to the final code? *

❏Yes

❏No

A3. Were you able to localise alternative texts (for non-text content such as images)? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A4. Were you able to localise bypass blocks elements? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A5. Were you able to translate the title of the page? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A6. Were you able to localise the links? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A7. Were you able to adapt the language attribute of the page?

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A8. Were you able to adapt the language attribute of the parts in French? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A9. Were you able to localise the 'on input' elements, such as the submit button? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A10. Were you able to localise the error messages? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A11. Were you able to localise the instructions embedded in the code, such as ARIA- attributes? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

A12. When working in the tool's editor, did the tool offer useful information about the features' context (for example, through showing the document structure or tags)? Please specify. *

Please write your answer here:

A13. Were any error messages displayed? *

❏Yes

❏No

A14. If yes, were they related to accessibility issues? Please specify to which accessibility feature they were related.

Please write your answer here:

A15. Do you think that using this CAT tool could have a positive or negative impact when localising accessibility features? Please specify. *

Please write your answer here:

MemoQ v8.7

B1. Were you successful in producing an accessible target file? *

❏Yes

❏No

B2. Did you have to have to make any manual modification to the final code? *

❏Yes

❏No

B3. Were you able to localise alternative texts (for non-text content such as images)? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B4. Were you able to localise bypass blocks elements? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B5. Were you able to translate the title of the page? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B6. Were you able to localise the links? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B7. Were you able to adapt the language attribute of the page?

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B8. Were you able to adapt the language attribute of the parts in French? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B9. Were you able to localise the 'on input' elements, such as the submit button? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B10. Were you able to localise the error messages? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B11. Were you able to localise the instructions embedded in the code, such as ARIA- attributes? *

❏Yes, in the tool's editor interface.

❏Yes, through manual changes.

❏No.

❏I don't know.

B12. When working in the tool's editor, did the tool offer useful information about the features' context (for example, through showing the document structure or tags)? Please specify. *

Please write your answer here:

B13. Were any error messages displayed? *

❏Yes

114

❏No

B14. If yes, were they related to accessibility issues? Please specify to which accessibility feature they were related.

Please write your answer here:

B15. Do you think that using this CAT tool could have a positive or negative impact when localising accessibility features? Please specify. *

Please write your answer here:

Thank you for your time and your collaboration!

Isotta Pacati

Appendix G.

Detailed results per participant

P01	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	0	0
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	0	0
TOTAL	76	76	76	76
%	95%	95%	95%	95%

P02	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	0	0
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	0	0
TOTAL	76	76	76	76
%	95%	95%	95%	95%

P03	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	1	0	1	1
Language of Parts	0	0	2	2
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	30	30
TOTAL	77	76	109	109
%	96%	95%	136%	136%

P04	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	0	0
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	1	1
TOTAL	76	76	77	77
%	95%	95%	96%	96%

P05	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	0	0
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instruction	0	0	0	0
TOTAL	76	76	76	76
%	95%	95%	95%	95%

P06	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	4	5
Link Purpose	46	46	46	46
Language of Page	0	0	1	1
Language of Parts	0	0	2	2
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	1	1
TOTAL	76	76	83	84
%	95%	95%	104%	105%

P07	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	1	1
Language of Parts	0	0	2	2
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	1	1
TOTAL	76	76	80	80
%	95%	95%	100%	100%

P08	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	1	1
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	1	1
TOTAL	76	76	78	78
%	95%	95%	98%	98%

P09	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	1	1
Link Purpose	46	46	46	46
Language of Page	0	0	0	0
Language of Parts	0	0	0	0
On Input	3	3	3	3
Error Identification	5	5	5	5
Labels or Instructions	0	0	0	0
TOTAL	76	76	76	76
%	95%	95%	95%	95%

P10	TRADOS	MEMOQ	TRADOS 1	MEMOQ 1
Non-text Content	18	18	18	18
Bypass Blocks	3	3	3	3
Page Titled	1	1	6	6
Link Purpose	46	46	46	46
Language of Page	1	0	1	1
Language of Parts	0	0	2	2
On Input	3	3	10	10
Error Identification	5	5	5	5
Labels or Instructions	1	0	22	22
TOTAL	78	76	113	113
%	97.50%	95%	141%	141%