Journal of Universal Computer Science, vol. 14, no. 16 (2008), 2699-2719
submitted: 9/7/07, accepted: 15/10/07, appeared: 28/8/08 © J.UCS

Supporting the Development of Accessible Web Applications

Myriam Arrue
(University of the Basque Country-Euskal Herriko Unibertsitatea
Laboratory of HCI for Special Needs
Manuel Lardizabal 1, 20018 Donostia, Spain
myriam@si.ehu.es)

Markel Vigo
(University of the Basque Country-Euskal Herriko Unibertsitatea
Laboratory of HCI for Special Needs
Manuel Lardizabal 1, 20018 Donostia, Spain
markel@si.ehu.es)

Julio Abascal
(University of the Basque Country-Euskal Herriko Unibertsitatea
Laboratory of HCI for Special Needs
Manuel Lardizabal 1, 20018 Donostia, Spain
julio.abascal@ehu.es)

Abstract: The aim of this paper is to review the best known methodologies for web applications development as well as the existing supporting tools and techniques from an accessibility-centric perspective. To this end, a number of development methodologies with their respective characteristics are described: model-based methodologies, user-centred processes, usability engineering methodologies and accessibility engineering methodologies. Some of these methodologies are provided with specific supporting tools which facilitate the accomplishment of specified tasks. However, there are methodologies which are not supported by specific tools. Therefore, web developers must deal with diverse tools in order to perform the corresponding activities. In this context, the development of accessible web applications is even more difficult. This paper concludes that there is not currently a holistic development framework to be used throughout the whole development process. Our contribution relies on a set of tools that support the different phases of the process. Since these tools are developed upon a common methodological basis, a high rate of interoperability is obtained. This cohesion allows their integration in a comprehensive framework so that the development of accessible web applications is facilitated.

Keywords: Web accessibility, Web engineering, Web applications, development process, development supporting tools
Categories: H.5.2, H.5.4, H.3.5

1 Introduction

Web applications play an increasingly important role in carrying out everyday activities in our life. The amount of information and services provided by web applications is rising vertiginously. In this sense, the World Wide Web (WWW) offers people with disabilities a plethora of services which have the potential to make them less dependent and more autonomous. On-line shopping, banking, education, etc., theoretically facilitate these daily tasks. However, due to inadequate design practices the WWW remains inaccessible for most users with physical, sensory or cognitive disabilities. Applying Universal Design principles in the development of web applications is essential in order to ensure access for all types of users.

In recent years, the web applications development process has changed considerably. Initially, web applications consisted of a limited number of web pages, most of them static, with an informative or advertising aim. They were often developed by people with minimal experience in software development and the process was not led by any appropriate methodology. As a result, low quality web applications were deployed [Murugesan, 02]. In recent years, web applications have become more complex and nowadays they integrate different technologies. In addition, they cover diverse activities and can be classified in different categories based on their functionality: informational, interactive, transactional, workflow oriented, collaborative work environments, online communities, portal-oriented, ubiquitous and semantic web applications [Murugesan, 05] [Kappel, 06]. The following table, Table 1, shows examples of each type of web application.

Functionality/Category	**Examples**
Informational	Online newspapers, product catalogues, newsletters, manuals, reports, online books, etc.
Interactive	Registration forms, online games, etc.
Transactional	Online shops, online banking, travel agencies, etc.
Workflow oriented	Online planning and scheduling, inventory management, status monitoring, etc.
Collaborative work environments	Distributed authoring tools, collaborative design tools, etc.
Online communities	Discussion groups, online auctions, etc.
Portal-oriented applications	Community portals, online shopping malls, business portals, etc.
Ubiquitous applications	Customised services, location-aware services, Multi-platform delivery, etc.
Semantic Web applications	Recommendation systems, syndication, knowledge management systems, etc.

Table 1: Examples of each type of web application.

Figure 1 shows different types of web applications according to their degree of complexity and their development history. This figure illustrates that in general there is a correlation between the chronology of development and the degree of complexity. One exception is provided by portal-oriented applications, which present a lower level of complexity than collaborative work environments even though they are more recent. In addition, it has to be noted that a web application may evolve to different types throughout its lifecycle as new functionalities are added. Commonly, in the initial stages of the development process a simple web application is implemented which is then refined in subsequent stages by accommodating more functionality until a more complex web application is developed.

Therefore, although the WWW was initially designed as an information media, it has become an application media in the last few years [Ginige, 01]. Companies are also displaying a growing tendency to use web applications in their management processes [Hoffman, 05]. In this sense, previous stand-alone business applications are evolving into light web applications which have proven to be more manageable and easier to distribute.

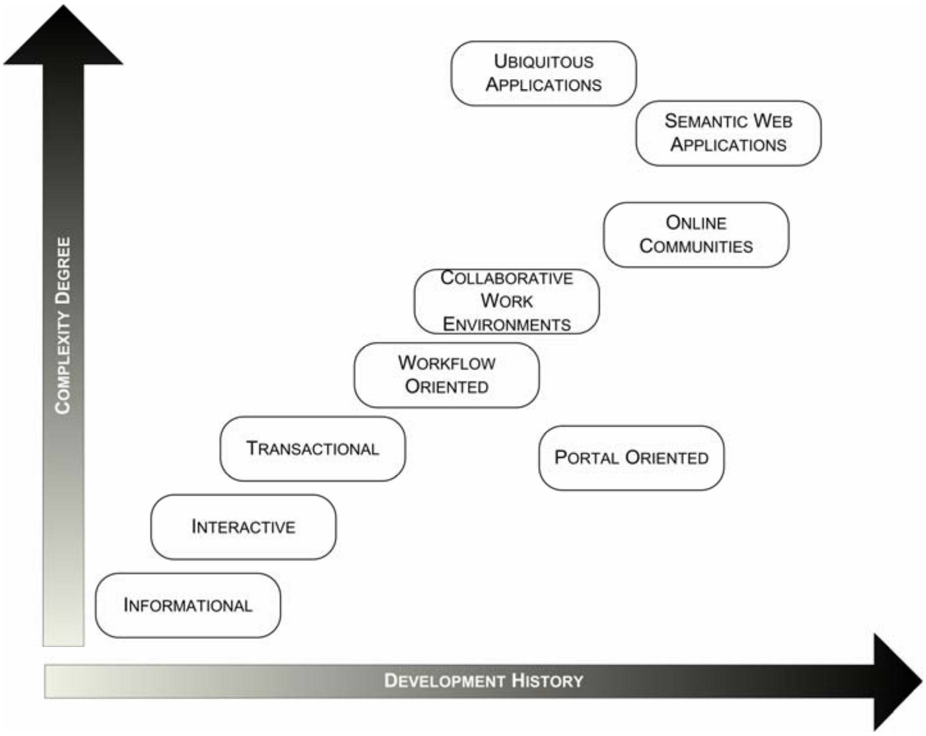


Figure 1: Classification of web application types according to their development history and degree of complexity [Kappel, 06].

Consequently, web applications development has changed from merely being a hypertext-based interface design process to a much more complex task which involves different activities such as planning, information architecture design, evaluation, quality assurance, system performance evaluation, maintenance, updates management and so on. Applying methodical, systematic development processes in order to guarantee the development of reliable, efficient, maintainable and secure web applications is of paramount importance. However, the development methodologies defined in the area of software engineering are not directly applicable to web applications, due to their specific characteristics [Mendes, 06]. It is in this context that the Web Engineering discipline has arisen. Its objective is to define appropriate techniques and methodologies to satisfy the needs of the web applications development process [Ginige, 01].

The aim of this paper is to highlight the most significant methodologies created for the development of accessible web applications as well as the existing techniques and tools for supporting the developers throughout the process. In addition, it presents specific tools and methods we have developed in order to support developers in performing tasks with regard to accessibility. These tools and methods may be integrated in commercial development frameworks to facilitate the implementation of accessible web applications.

2 Universal Accessibility

According to Brajnik [Brajnik, 00], web applications are interactive software systems which interact at least with two types of user: the end-users and developers. The objective of end-users when accessing these applications is to perform specific tasks in a satisfactory way. The developers access the web applications in order to perform updates and maintenance tasks. Therefore the users of web applications can be classified according to different factors:

- *The objectives and the tasks they try to perform*. For example, searching for information, buying a product, etc.
- *Users' context.* The users' behaviour is determined by their cultural level, interaction language, experience accessing the Web, etc.
- *Used technology*. Users interact with web applications through a technology layer transparent to web developers. In order to enable the interaction, different tools and devices, browsers, protocols, plug-ins, operating systems, connections of different characteristics, assistive technology, etc. may be used.

Several laws have been enacted all around the world[1] so that agencies supported by public funding would make their sites accessible to people with disabilities and the elderly. However, the remainder of websites are not required to meet Universal Access principles and because of that the WWW remains inaccessible to these people. A number of facts may encourage companies to develop accessible sites:

- In some countries people with disabilities reaches 20% of the population. Thus, making a website accessible could considerably increase its potential users and therefore raise the profits of a company.
- Accessible sites get a higher ranking in search engines [Pemberton, 03].
- It will positively affect company image.
- It is ethically the right thing to do.

These are some of the reasons why the "Universal Access" concept is becoming an extremely significant factor for the current Information Society. If the main objective is the methodical development and maintenance of accessible web

[1] http://www.w3.org/WAI/Policy/

applications, accessibility issues should be included in the methodologies defined in the area of Web Engineering.

Besides governmental efforts, other initiatives have also fostered web accessibility awareness. One of the most proactive initiatives is the Web Accessibility Initiative (WAI)[2] that was set up by the World Wide Web Consortium. This initiative published the well-known Web Content Accessibility Guidelines (WCAG)[3] which is the most universally accepted and established set of guidelines for developing and evaluating web content accessibility. In addition, several accessibility evaluation and repair tools have been developed.

Even though all these efforts are extremely useful for producing accessible web applications, they have proven not to be sufficient in order to achieve the Universal Access objective. Unfortunately, many websites required to be accessible by law are still not accessible. In 2004, Lazar, Dudley-Sponaugle and Greenidge [Lazar, 04] carried out a survey in order to gather web developers' perceptions about web accessibility. According to this study, web developers require methodologies which incorporate web accessibility issues throughout the entire development process. In the same study, confusing accessibility guidelines and lack of adequate software tools are two of the reasons given by web developers for the current low accessibility level of web applications.

In fact, a large amount of web accessibility guidelines have been developed recently. Nowadays, in addition to general purpose guidelines such as WCAG or Section 508, other sets of guidelines related to specific application type (informational, transactional, etc.), specific users' characteristics (elderly, teenage people, deaf, etc.) and accessing devices (mobile devices, etc.) are available. As a result, web developers may be confused when trying to deal with all this information defined so heterogeneously. We can conclude that even if laws, guidelines and tools have successfully spread web accessibility awareness, they are not enough to achieve accessible websites.

3 Web Applications Development Process

According to Sommerville [Sommerville, 92] a software application based on a correct development methodology should satisfy four key requirements: easy maintenance, reliability, efficiency and an appropriate user interface. The last property refers to the design of the interface according to the abilities of future application users. Diverse development methodologies have been defined in order to facilitate the development of software applications which satisfy these basic properties. They give guidance for planning, organising, coordinating and managing software development activities.

The characteristics of the software application to be developed, such as size, complexity and specific features, as well as the temporal restrictions, should be taken into account when selecting a specific process model for the current development.

As mentioned above, web applications initially consisted of a limited number of web pages, most of them static, with an informative or advertising aim. They were often

[2] http://www.w3.org/WAI/

[3] http://www.w3.org/WAI/intro/wcag.php

developed by people with minimum experience in software development. Therefore, the development process was not based on any appropriate methodology. This has led to the development of poor quality web applications which are difficult to maintain [Murugesan, 02].

Therefore, the need for methodical, systematic development processes is obvious, in order to guarantee the development of reliable, efficient, easy to maintain and secure web applications. However, the development methodologies defined in the area of software engineering are not directly applicable to web applications, due to their specific characteristics [Mendes, 06] [Kappel, 06]. Some of these specific characteristics of web applications are the following:

- Web applications developers are not always experts in the area.
- Technologies integrated in web applications are in constant evolution.
- There is a need for integrating different technologies and systems.
- End-users' characteristics are often unknown.
- Usage of different devices to access the web applications.
- Complex maintenance process due to frequent updates.
- Difficult estimation of end-users number.
- Abilities, knowledge and preferences of end-users are heterogeneous.
- Internationalisation of web applications in terms of cultural and linguistic differences has to be considered due to global access of the Web.
- Stability of the system is crucial as it is supposed to be permanently operable.

The following sections describe different development methods proposed in the Web Engineering field.

3.1 Development Methodologies Based on Models

According to Fraternali [Fraternali, 99], a web application is defined by three major design dimensions:

- *Structure* describes the different parts that form the web application and their semantic relationships.
- *Navigation* comprising the facilities for accessing the content and browsing in the application.
- *Presentation* describes the way content and navigation mechanisms are presented to the user.

The development methodologies based on models provide the necessary primitives and mechanisms for specifying structural, navigational and presentational high-level views by abstracting them from any architectural issue. Therefore, these methodologies are based on the specification of these techniques.

The traditional hypermedia development methodologies are based on models. The soundest methodologies are the Hypermedia Design Model (HDM) [Garzotto, 93], Relationship Management Methodology (RMM) [Isakowitz, 95], Object Oriented HDM (OOHDM) [Schwabe, 95], WebML [Ceri, 00] and Autoweb [Fraternali, 00]. However, none of the previously mentioned web engineering methodologies integrate the necessary web accessibility related activities. Montero, Díaz and Aedo [Montero, 03] propose a framework for the analysis and comparison of the hypermedia development methodologies. This framework lists the requirements which methodologies have to fulfil and analyses the most widespread methodologies. The requirements are the following:

- R_1: Provide the designer with artefacts to specify system requirements.
- R_2: Provide software support tools to help in the systems development process.
- R_3: Model the different types of users.
- R_4: Provide means to describe the interactive behaviour of hypermedia systems.
- R_5: Make possible the evaluation of system utility.

The results of this analysis are shown in the following table, Table 2.

Requirements	**HDM**	**RMM**	**OOHDM**	**Autoweb**	**WebML**
R_1	P	P	P	P	P
R_2	C	P	P	C	C
R_3	N	N	C	N	C
R_4	C	N	C	N	C
R_5	C	C	N	N	N

Table 2: Results of the analysis carried out in order to determine the fulfilment of the requirements R_1-R_5. Notation: P stands for Partially fulfilled requirement, C stands for Completely fulfilled requirement and N stands for Not fulfilled requirement [Montero, 03].

According to the analysis carried out, none of the methodologies provide the necessary mechanisms for specifying the non-functional requirements such as accessibility, usability, efficiency, etc. Most of the methodologies analysed are supported by software tools; and three of them, HDM, Autoweb and WebML, completely fulfilled this requirement. However, these tools do not provide the necessary functionalities for the maintenance phase. The majority of tools provide functionalities that may be applied from conceptualisation to the implementation phase. OOHDM and WebML are the only methodologies which provide mechanisms for user modelling and RMM and Autoweb do not provide any support for describing the interactive behaviour of the system. Finally, this study analyses the possibility of performing system evaluations. HDM and RMM are the only methodologies that support this task even if the criteria of the evaluations are based on design features [Garzotto, 95].

3.2 User-Centred Development Methodologies

The objective of User-Centred Development (UCD) methodologies is to develop user interfaces which can be used by all types of users regardless of their abilities. Therefore, these methodologies should consider the users' characteristics, context of use, tasks to perform, etc.

Initially, the UCD methodologies were based on the use of specific usability methods and techniques in isolation. Nowadays, several frameworks for integrating these methods and techniques have been developed [Stephanidis, 01].

The best known framework is proposed in the ISO 13407: *Human-centred design processes for interactive systems* [ISO/IEC, 99] standard, developed in 1999. The objective of this standard is to provide a guide for the development of usable interactive systems by incorporating the user-centred design into their lifecycle. The following figure, Figure 2, shows the development process proposed in this standard.

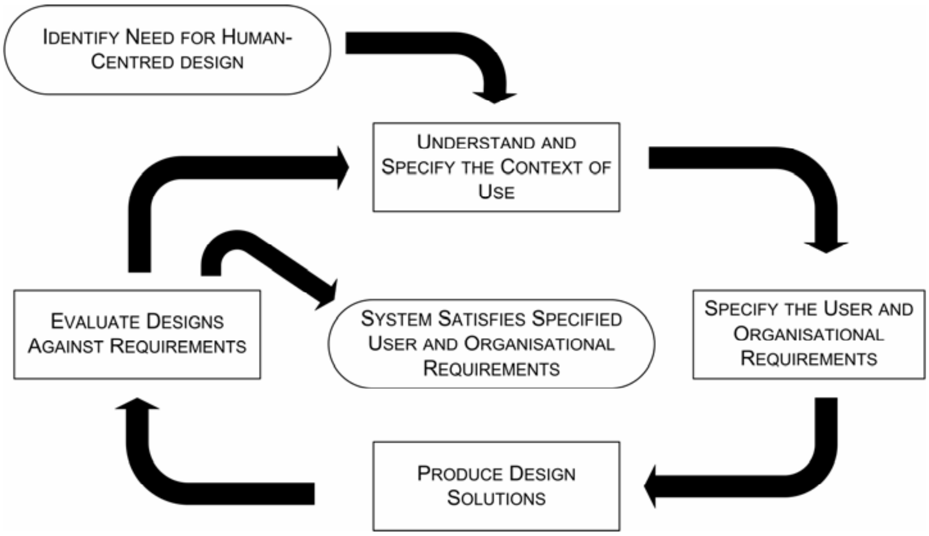

IDENTIFY NEED FOR HUMAN-CENTRED DESIGN
UNDERSTAND AND SPECIFY THE CONTEXT OF USE
SPECIFY THE USER AND ORGANISATIONAL REQUIREMENTS
PRODUCE DESIGN SOLUTIONS
EVALUATE DESIGNS AGAINST REQUIREMENTS
SYSTEM SATISFIES SPECIFIED USER AND ORGANISATIONAL REQUIREMENTS

Figure 2: Web Applications development process proposed in ISO 13407 standard.

According to Jokela, Iivari, Matero and Karukka [Jokela, 03] this standard can not be considered as a comprehensive methodology as it does not provide details about the methods and techniques to apply in order to determine the effectiveness, efficiency and end-user satisfaction when using the developed system. In this context, a comprehensive study of the methods and techniques available can be found in [Maguire, 01].

Web Site Design Method (WSDM) [De Troyer, 98] [De Troyer, 05] is a user-centred development methodology based on models. Its salient point is an initial user analysis in order to obtain the characteristics of different groups of users. Subsequently, the information is modelled according to these characteristics. Therefore, more usable interfaces will be obtained as they will contain functionalities for interface personalisation.

3.3 Usability Engineering

The principal objective of Usability Engineering is to define development processes which incorporate activities regarding usability. According to Nielsen, the objective of Usability Engineering is to build bridges, provide links and create connections between the users' abilities and the possibilities that computers offer [Nielsen, 93].

One of the first established development processes in the Usability Engineering area was the model proposed by Nielsen. It lists the necessary activities that a comprehensive Usability Engineering development methodology should incorporate. In addition, it specifies several usability evaluation techniques and methods.

The Usability Engineering Lifecycle was proposed in 1999 by Mayhew [Mayhew, 99]. This methodology defines three basic phases for the development process: requirements analysis, design/development/testing and installation. The user interface design is the key step in this methodology. The prototyping techniques and usability evaluation are integrated in the Usability Engineering Lifecycle.

Although the two methodologies described above are the most referenced ones there are other ones such as MPIu+a [Granollers, 04]. This methodology attempts to integrate aspects from software engineering, human-computer interaction and usability engineering. The principal components of the proposed process model are: software engineering development process, prototyping technique and evaluation.

3.4 Accessibility Engineering

The Unified Web Evaluation Methodology (UWEM) [Velleman, 06] is a methodology specifically defined for evaluation of web application accessibility proposed by the Web Accessibility Benchmarking Cluster (WAB Cluster) [http://www.wabcluster.org/] which is conformed by the collaboration of several European research projects: BenToWeb, EIAO and Support-EAM.

The objective of UWEM, which is focused on the evaluation stage, is to facilitate the compatibility and coherence of automatic accessibility evaluation and monitoring tools. It consists of several principles and best practices which support both manual evaluations by experts and automatic evaluations. Since the methodology is based on WCAG 1.0 set of guidelines, it aims at unifying the interpretations given to each guideline.

Other methodologies have been defined in Accessibility Engineering area. Some of them attempt to incorporate accessibility into user-centred development. The one presented by Henry and Grossnickle [Henry, 04] is a methodology which considers accessibility as a subset of usability. Therefore, accessible development techniques are integrated into each phase of the development process. Stephanidis, Akoumianakis, Sfyrakis, and Paramythis [Stephanidis, 98] propose another methodology for integrating accessibility into user-centred development is presented. This study proposes a conceptual framework independent from any particular technology for incorporating accessibility into the user interface lifecycle. For this purpose, it determines some process-oriented accessibility guidelines which extend the user-centred user interface development.

A methodology integrating accessibility in a model-based development is presented in [Plessers, 05]. This methodology relies on the automatic annotation of objects in web pages with semantic knowledge so that screen readers can render the content more efficiently. The semantic knowledge is represented in the Web Authoring for Accessibility (WAfA) ontology [Yesilada, 04]. This process will be automatic and extends the previously mentioned WSDM model-based methodology.

Another model-based methodology for the integration of accessibility aspects into the development process is described in [Jeschke, 06]. In this case, the objective is to analyse the way accessibility aspects could be integrated in the development of e-learning platforms.

Nevertheless, these methodologies do not consider web accessibility as a whole. Many of them focus on specific stages of the lifecycle (such as the evaluation phase) whereas others focus on users' specific disabilities (such as visually impaired users).

4 Tools for Supporting the Development Process

The methodologies described in the previous section are difficult to apply by developers without the help of supporting tools and techniques. This section is dedicated to analysing the existing tools for web application development.

An exhaustive study about the existing tools for supporting web applications development can be found in [Fraternali, 99]. According to this study these tools can be classified into six different types, based on their functionalities:

1. Visual editors and site managers
2. Web-enabled hypermedia authoring tools
3. Web-DBPL (Data Base Programming Language) integrators
4. Web form editors, report writers, and database publishing wizards
5. Multi-paradigm tools
6. Model-driven application generators

None of these tools covers the development process of web applications from a holistic approach since they focus on specific stages of the process. In Table 3, the functionalities supported by each type of tool are specified.

4.1 Tools and Techniques to Support Web Accessibility

Although some of the tools mentioned have functionalities related to web accessibility, more specific tools and techniques for facilitating the development of accessible web applications have been developed in recent years.

One complex aspect that developers have to deal with is the management of accessibility guidelines throughout the development process. In fact, the extensive amount of information about web accessibility in terms of best practices, techniques and sets of recently developed guidelines makes it difficult to perform activities such as:

- Search for the sets of guidelines which are significant for the current development.
- Select the most appropriate sets of guidelines.
- Remove guideline overlaps and resolve conflicts.
- Verify the coherence of the selected sets of guidelines.
- Analyse the applicability of the selected guidelines in the current development.
- Develop directly applicable design rules from the selected guidelines.
- Plan and perform frequent accessibility evaluations based on the selected sets of guidelines during the development process.

Several Guidelines Management Tools have been developed in order to facilitate the development process of accessible web applications. SIERRA [Vanderdonckt, 95] is one of the first approaches for managing usability knowledge by a software tool. However, this tool does not support any evaluation process. Sherlock [Grammenos, 00] manages usability guidelines by a client-server system and evaluates automatically only some of the defined guidelines. Another system, called GUIDE [Henninger, 00], manages and stores usability guidelines for their subsequent application in a particular environment. Nevertheless, none of these approaches supports the whole development process.

	1	**2**	**3**	**4**	**5**	**6**
A						Facilities for model generation
D	Presentation design	Navigation design	Structural design		Presentation design	Structural, navigational, presentational model
I	Code generation	Code generation, Data Base connectivity	DB queries & web pages integrating facilities	Code generation	Code generation, DB content publishing	Code generation, DB generation, connectivity
E				Code debugging		
M	Web site management facilities		Content maintenance	Version control, configuration management	Web site management facilities Content maintenance	

Table 3: Functionalities implemented in each type of tool for performing activities of specific development process phases. Notation: A stands for Analysis Phase, D stands for Design Phase, I stands for Implementation Phase, E stands for Evaluation Phase and M stands for Maintenance Phase.

More recent approaches, such as the ones proposed by Mariage and Vanderdonckt [Mariage, 04] and Leporini, Paternò and Scorcia [Leporini, 06] are useful throughout the web applications development process, including the evaluation stage. Both aim at simplifying the interaction with accessibility guidelines using graphical interfaces. Unfortunately, both are stand-alone applications which have some drawbacks compared to a web application. Moreover, the guideline formats used by these applications have not proven to be developed from the analysis of the state of art of different types of guideline sets. Therefore, some guidelines may not be adequately evaluated or specified.

In addition, a large number of automatic tools for web accessibility evaluation have been developed [http://www.w3.org/WAI/ER/tools/] [Ivory, 03]. Most of them evaluate predefined sets of general purpose accessibility guidelines such as WCAG 1.0, Section 508, etc. In addition, there are few tools which also allow evaluating more specific guideline sets in relation to the type of web application or users' characteristics [Vanderdonckt, 05] [Leporini, 06].

When developing high quality web applications the management of Quality Assurance processes is essential. This implies the necessity of applying metrics, methods and quality models in the development process. Web accessibility has also to be considered within this activity in order to develop accessible web applications.

Some quality models such as the 2QCV3Q [Mich, 03] and the WebQEM [Olsina, 02] have been defined. The attributes of web applications and the necessary metrics for their evaluation are enumerated in these models. However, none of these models consider accessibility as an essential property of web applications, since it is included as an attribute of other properties.

Therefore, evaluating methods and metrics for measuring the accessibility are necessary for quality assessment and analysing the evolution of the accessibility. In the last few years, several quantitative metrics for measuring web accessibility have been defined. Sullivan & Matson [Sullivan, 00] evaluate only eight checkpoints from WCAG 1.0. The "failure-rate" is the ratio between potential errors and real errors. Hackett, Parmanto and Zeng [Hackett, 04] proposed the WAB formula (Web Accessibility Barrier). This formula uses as input parameters the total number of pages of a website, total accessibility errors and potential errors in a web page, and error priority. Bühler, Heck, Perlick, Nietzio and Ullveit-Moe [Bühler, 06] propose a novel approach in order to adapt measurement to different disability groups. However, these metrics are still in the development stage until better results are obtained. As far as usability metrics for the visually impaired are concerned, Fukuda, Saito, Takagi and Asakawa [Fukuda, 05] automatically measure the *navigability* and *listenability* of a web application.

5 A Framework for Accessible Web Applications Development

The previously presented tools and methodologies for accessible web applications development do not cover all the development process since they just focus on specific phases of the lifecycle. Most of the existing tools focus on evaluation procedures and therefore their integration into a development environment is a partial, incomplete solution. Web developers are not provided with any unified interface to easily integrate accessibility related activities in development environments. Therefore, they are forced to deal with several interfaces and interpret several output formats which might not be interoperable. These constraints lead web developers to forget accessibility issues throughout the development process and consider them only when the web application is in the latter stages of implementation. Thus, repairing accessibility errors in these later iterations requires major changes and higher costs.

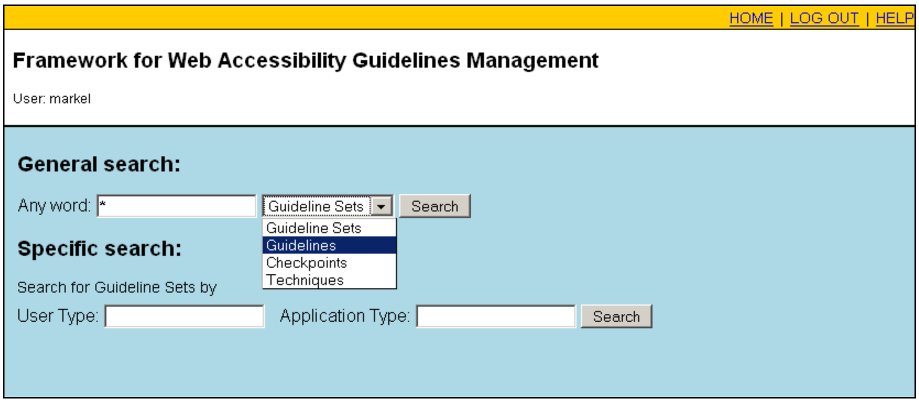
We have developed several tools and techniques which cover the different activities in the lifecycle of web applications. These tools can be easily integrated with other development tools to form a comprehensive framework for accessible web applications development.

In the following sections, we highlight the activities which have to be carried out in relation to web accessibility, and the tools, techniques, methods and models which will be useful in each stage of the lifecycle are also included. In addition, we meet some of these requirements by means of prototypes and techniques that implement sound methods.

5.1 Analysis Phase

In this stage, a user-centred approach concerning the analysis of end-users' specific characteristics is the main activity to be carried out regarding accessibility. In some cases, these features are known, for example, when developing a web application for an intranet. However, there are other scenarios where the objective will be the development of a web application which can be accessed by all groups of users regardless of their abilities and characteristics. In these cases, it is essential to elaborate user profiles which contain the necessary characteristics so that as many as possible different groups of users are included. However, as stated by Abascal and Nicolle [Abascal, 05], the broad diversity of users and disabilities makes it difficult to include all potential users. "Universal Design" guidelines and techniques have to be considered in order to create user profiles which do not exclude any group of users. Therefore, it is essential to perform an exhaustive analysis of the existing sets of guidelines in order to select the most appropriate, according to the type of web application to be developed and end-users' characteristics. Activities related to accessibility are the following:

- Analysis of end-users' features and characteristics of the application.
- Bear in mind the "Design for All" paradigm to avoid excluding user profiles.
- Analysis and selection of guideline sets to be applied in the development.


Framework for Web Accessibility Guidelines Management

User: markel

General search:

Any word: * Guideline Sets

Specific search:

Search for Guideline Sets by

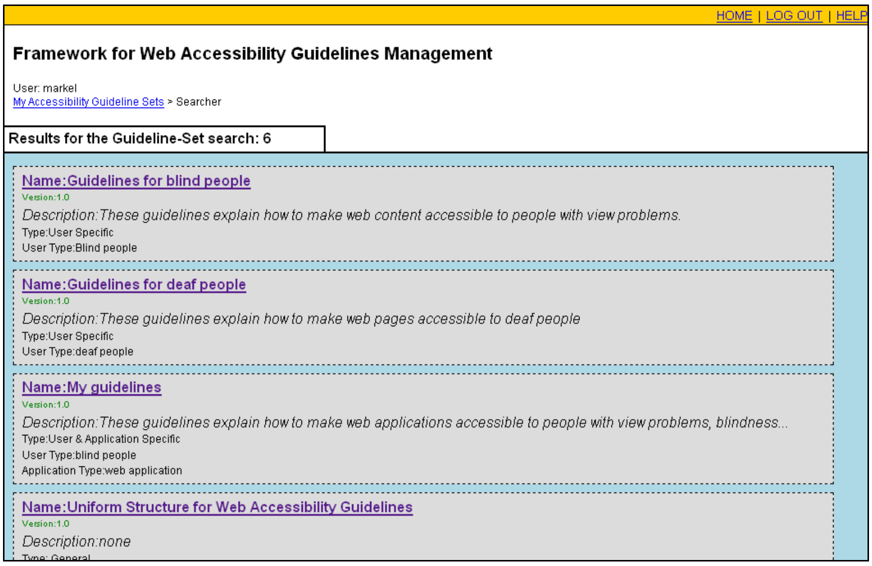
User Type: Application Type:

Figure 3: Guidelines, checkpoints and techniques search.

The process of seeking adequate guidelines is of great significance, as the selected sets will be used for the entire development process. Since the sets of guidelines could be published in diverse formats, their automatic manipulation is a challenging task. In this sense, we have designed an XML-based uniform language for the representation of guidelines[4] [Arrue, 07a]. In addition, we have developed a framework for guidelines management [Arrue, 07b]. One of its main features consists of a guidelines search tool which allows queries to be performed depending on end-user features or application type. Thus, it can be used as an online guidelines

[4] Further information can be obtained in: http://sipt07.si.ehu.es/evalaccess3/ugl.xsd

repository so that knowledge about web accessibility can be easily retrieved and shared among developers. These guidelines sets will subsequently be useful in the design stage and for evaluation purposes. Figures 3 and 4 are screenshots of the accessibility guidelines search interfaces.



Framework for Web Accessibility Guidelines Management

User: markel
My Accessibility Guideline Sets > Searcher

Results for the Guideline-Set search: 6

Name:Guidelines for blind people
Version:1.0
Description:These guidelines explain how to make web content accessible to people with view problems.
Type:User Specific
User Type:Blind people

Name:Guidelines for deaf people
Version:1.0
Description:These guidelines explain how to make web pages accessible to deaf people
Type:User Specific
User Type:deaf people

Name:My guidelines
Version:1.0
Description:These guidelines explain how to make web applications accessible to people with view problems, blindness...
Type:User & Application Specific
User Type:blind people
Application Type:web application

Name:Uniform Structure for Web Accessibility Guidelines
Version:1.0
Description:none
Type: General

Figure 4: Guidelines search result.

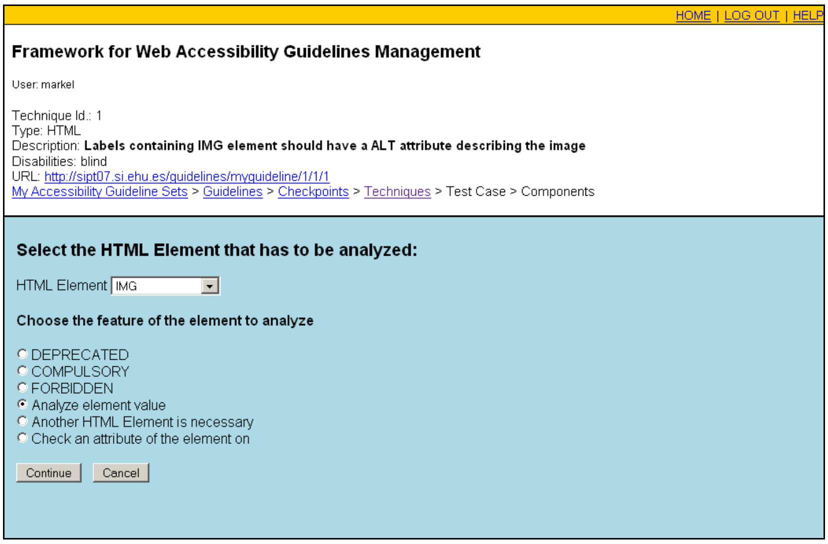
5.2 Design Phase

It is essential to use appropriate tools in order to model user tasks defined in the analysis phase as well as the specific features of the web application and characteristics of end-users. The navigation mechanism and movement between the different web pages are defined in this stage. For this purpose, it is necessary to deal with the different cognitive models of the groups of users being considered. In early iterations the interface is a vague approximation to what is intended to be developed, and tends to be a preliminary sketch usually drawn on paper Afterwards, these first drafts evolve into real web scenarios where the design is carried out using design authoring tools.

In some cases, the selected guidelines have to be interpreted by means of certain techniques in order to be applied. These techniques have to be identified, for example the TWCAG techniques [http://www.w3.org/TR/WCAG10-TECHS/], which implement the WCAG set of guidelines. We have developed a web application for techniques definition; Figure 5 shows a screenshot. Developers will be able to include their own interpretations of guidelines and store them for evaluation or sharing purposes. Since many web developers find guidelines definitions ambiguous and confusing, this tool allows sharing of the interpretations of guidelines among the developers' community and facilitates the communication between those with more expertise and new developers. Activities related to accessibility are the following:

- Interpretation of selected guidelines.

- Identification of techniques to apply guidelines correctly.
- Design and development of appropriate navigation schemas based on selected cognitive models.



Framework for Web Accessibility Guidelines Management

User: markel

Technique Id.: 1
Type: HTML
Description: **Labels containing IMG element should have a ALT attribute describing the image**
Disabilities: blind
URL: http://sipt07.si.ehu.es/guidelines/myguideline/1/1/1
My Accessibility Guideline Sets > Guidelines > Checkpoints > Techniques > Test Case > Components

Select the HTML Element that has to be analyzed:

HTML Element IMG

Choose the feature of the element to analyze

○ DEPRECATED
○ COMPULSORY
○ FORBIDDEN
● Analyze element value
○ Another HTML Element is necessary
○ Check an attribute of the element on

Figure 5: Including design techniques for later evaluation purposes.

5.3 Implementation Phase

The user interface should be developed in order to efficiently satisfy the needs of different groups of end-users. In this sense, it is essential to consider all the alternative content that should be included. This would guarantee that all users will be able to access most of the content. Activities related to accessibility are the following:

- Application development according to the selected guidelines.
- Usage of appropriate authoring tools.
- Implementation of a single flexible user interface.
- Incorporation of all the necessary alternative content.

The architecture of the implemented tools facilitates their integration and their interoperability with other applications like authoring tools. The knowledge stored in the repositories of guidelines and the results obtained by these tools are XML-based.

5.4 Evaluation Phase

The fulfilment of all the specifications defined in the analysis stage and the quality level of the final product are verified in this stage. As far as web accessibility is concerned, the evaluations performed in this stage will determine whether the web application fulfils the specified accessibility level. Therefore, diverse accessibility evaluations have to be performed in this stage in order to detect any possible barrier and fix it. Performing a comprehensive evaluation implies combining diverse types of evaluations:

- Automatic evaluation with tools: this is a preliminary test stage aiming to remove the first and most "evident" obstacles. "Evident" means those errors automatically testable with the help of tools. According to Lang [Lang, 03], this evaluation method presents diverse advantages in terms of costs and efficiency, as automatic evaluation tools yield error reports in a short period of time. The aim of this evaluation is to clear up the content so that forthcoming evaluations with experts and users take less time and they can focus on other complex issues. An effective evaluation tool should be able to validate the fulfilment of most of the guidelines. Yet, nowadays it is a distant objective, since there is not enough research being done to evaluate some checkpoints such as the WCAG 1.0 14.1 checkpoint: "Use the clearest and simplest language appropriate for a site's content".
- Expert-driven manual evaluations: as abovementioned, the evaluation of some guidelines requires human judgment. Web accessibility experts perform evaluations based on heuristics in order to evaluate these kinds of guidelines. Performing the main tasks in a given web application and carrying out walkthroughs with different browsers, assistive technologies, devices, etc., is another way of testing. These evaluation methods allow the detection of accessibility barriers under different conditions of use [Brajnik, 06].
- Evaluations with users: this evaluation type is essential since it allows the detection of real accessibility barriers for users with specific characteristics. Selected users should cover the broader range of disabilities if a comprehensive evaluation is required. The evaluations coincide with tasks carried out with users in relation to the main functionalities of the web application. These evaluations take place in controlled environments such as specific laboratories where the experts can observe the actions of the users and gather information about the interaction following accepted usability evaluation techniques such as the ones described in Nielsen and Mack [Nielsen, 94] and Rubin [Rubin, 94]. However, results obtained from remote evaluations carried out in the users' normal environments can be also useful, as mentioned in [Petrie, 06]. All the problems detected should be analysed and fixed.

All these evaluations are complementary and necessary. If only automatic evaluations are carried out the fulfilment of several guidelines will not be checked and the required minimum accessibility level is seldom reached. On the other hand, evaluations with users also help in finding out usability barriers which accessibility guidelines and therefore automatic accessibility evaluation tools do not consider. Activities related to accessibility are the following:

- Accessibility evaluation with automatic tools, experts and disabled users.
- Evaluation of the quality of the web application.
- Documentation of all the detected errors.
- Repair detected errors.

Guideline sets and techniques defined and obtained in the previous stages have to be incorporated in flexible evaluation tools. We have developed an evaluation tool that can be easily integrated into other applications. Therefore, it can interact with the previously presented guidelines management tool in order to evaluate the guidelines retrieved from its repository [Abascal, 04].

5.5 Maintenance Phase

Due to the dynamic nature of the WWW, updates are frequent in web applications and the accessibility level and the quality tend to decrease. Nowadays, these updates are commonly managed by Content Management Systems (CMS). To our best knowledge, none of the existing commercial CMSs considers web accessibility issues. Therefore, it is essential to monitor the accessibility level of a web application. This stage could be understood as the accessibility monitoring stage since the evolution/involution of accessibility should be measured. Determining whether an update has increased or decreased the accessibility level of a web application is a complex task which has to be carried out in order to fix errors and maintain its accessibility and quality level. Activities related to accessibility are the following:

- Monitoring of the accessibility level of the web application.

We have defined web accessibility quantitative metrics [Arrue, 05] [Vigo, 07] which accurately measure the accessibility level of a web application in order to monitor its accessibility evolution. Due to the aforementioned flexibility and interoperability, these metrics can be automatically calculated by a tool which has been developed with this aim. Currently, we are integrating this feature into a monitoring tool which accurately computes the evolution of the accessibility level in web applications during their lifecycle.

6 Conclusions

In the last few years, many initiatives have been launched in order to foster web accessibility. These initiatives have promulgated the elaboration of a large amount of information related to web accessibility. However, web developers find it difficult to deal with all this information. Therefore, specific methodologies which guide designers developing accessible web applications are necessary. These methodologies should define the accessibility-related activities necessary and establish an ordered process for the development tasks.

In this context, many development methodologies have been defined, each with their respective characteristics: some of them are based on models; others are user-centred, etc. However, it is difficult to implement the defined methodologies if the appropriate supporting tools are not available.

This paper revised the most widespread methodologies for web applications development from an accessibility-centric perspective. In addition, it investigated the existing supporting tools. Some of the methodologies specify the techniques and tools adequate for carrying out the necessary tasks. However, there is no a comprehensive framework which supports the entire process of accessible web application development. Consequently, developers are forced to use diverse tools with different interfaces and information formats.

This paper proposed several tools that we have implemented to facilitate the development process. They are useful for performing specific accessibility-related activities and can be easily integrated into a more comprehensive development framework.

References

[Abascal, 04] Abascal, J., Arrue, M., Fajardo, I., Garay, N., Tomas, J. (2004). Use of Guidelines to automatically verify web accessibility. International Journal on Universal Access in the Information Society (UAIS), Vol. 3, No. 1, pp. 71-79.

[Abascal, 05] Abascal, J., Nicolle, C. (2005). Moving towards inclusive design guidelines for socially and ethically aware HCI. Interacting with Computers. Vol. 17, No. 5, pp. 484-505.

[Arrue, 05] Arrue, M., Vigo, M., Abascal, J. (2005). Quantitative Metrics for Web Accessibility Evaluation. In Proceedings of the ICWE 2005 Workshop on Web Metrics and Measurement. University of Wollongong School of IT and Computer Science.

[Arrue, 07a] Arrue M., Vigo M., Abascal, J. (2007). Including Heterogeneous Web Accessibility Guidelines in the Development Process. Engineering Interactive Systems 2007. IFIP WG 2.7/13.4, IFIP WG 13.2 and DSVIS Conferences. To be published in LNCS. Springer.

[Arrue, 07b] Arrue, M., Vigo, M., Aizpurua, A., Abascal, J. (2007). Accessibility Guidelines Management Framework. In C. Stephanidis (Ed.). Universal Access in HCI, Part III, HCII 2007. LNCS 4556. Springer, pp. 3-10.

[Brajnik, 00] Brajnik, G. (2000). Automatic web usability evaluation: what needs to be done? In Proc. Of 6th Human Factors and the Web Conference, June 2000, Austin, Texas.

[Brajnik, 06] Brajnik, G. (2006). Web Accessibility Testing: When the Method Is the Culprit. In K. Miesenberger et al (Eds.). Proceedings of Computers Helping People with Special Needs 2006. Lecture Notes in Computer Science 4061. Springer, pp 234-241.

[Bühler, 06] Bühler, C., Heck, H., Perlick, O., Nietzio, A., Ullveit-Moe, N. (2006). Interpreting Results from Large Scale Automatic Evaluation of Web Accessibility. In K. Miesenberger et al (Eds.). Computers Helping People with Special Needs 2006. Lecture Notes in Computer Science 4061. Springer, 184-191.

[Ceri, 00] Ceri, S., Fraternali, P., Bongio, A. (2000). Web modeling language (WebML): a modeling language for designing web sites. Computer Networks, Vol. 33 No. 1-6, pp. 137-157.

[De Troyer, 05] De Troyer, O., Casteleyn, S., Plessers, P. (2005). Using ORM to Model Web Systems. In On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops, International Workshop on Object-Role Modeling (ORM'05). Lecture Notes in Computer Science 3762. Springer, pp. 700-709.

[De Troyer, 98] De Troyer, O. And Leune, C. J. (1998). WSDM: a user centred design method for Web sites. Computer Networks and ISDN Systems, Vol. 30 No. 1-7, pp. 85-94.

[Fraternali, 00] Fraternali, P. and Paolini, P. (2000). Model-driven Development of Web Applications: The Autoweb System. ACM Transactions on Information Systems (TOIS), Vol. 18, No. 4, pp. 323-382.

[Fraternali, 99] Fraternali, P. (1999). Tools and Approaches for Developing Data-Intensive Web Applications: A Survey. ACM Computing Surveys, Vol. 31, No. 3, pp. 227-263.

[Fukuda, 05] Fukuda, K., Saito, S., Takagi, H., Asakawa, C. (2005). Proposing new metrics to evaluate web usability for the blind. In Extended Abstracts Proceedings of the 2005 Conference on Human Factors in Computing Systems, CHI 2005, pp. 1387-1390.

[Garzotto, 93] Garzotto, F., Paolini, P., Schwabe, D. (1993). HDM: A model-based approach to hypertext application design. ACM Transactions on Information Systems (TOIS), Vol. 11, No. 1, pp. 1-26.

[Garzotto, 95] Garzotto, F., Mainetti, L., Paolini, P. (1995). Hypermedia Design, Analysis, and Evaluation Issues. Communications of the ACM, Vol. 38, No. 8, pp. 74-86.

[Ginige, 01] Ginige, A. and Murugesan, S. (2001). Web engineering: A methodology for developing scalable, maintainable web applications. Cutter IT Journal. Vol. 14, No. 7, pp. 24-35.

[Ginige, 01] Ginige, A., Murugesan, S. (2001). Web Engineering – An introduction, Special Issue on Web Engineering, IEEE Multimedia. Vol. 8 No.1, pp. 14-18.

[Grammenos, 00] Grammenos, D., Akoumianakis, D., Stephanidis, C. (2000). Integrated support for working with guidelines: the Sherlock guideline management system. Interacting with Computers. Elsevier. Vol. 12, No. 3, pp. 281-311.

[Granollers, 04] Granollers, T. MPIu+a. Una Metodología que Integra la Ingeniería del Software, la Interacción Persona-Ordenador y la Accesibilidad en el Contexto de Equipos de Desarrollo Multidisciplinares. PhD dissertation. Universidad de Lleida. 2004.

[Hackett, 04] Hackett, S., Parmanto, B., Zeng, X. (2004). Accessibility of Internet websites through time. In Proceedings of 6th International ACM SIGACCESS Conference on Computers and Accessibility, pp. 32-39.

[Henninger, 00] Henninger, S. (2000). A methodology and tools for applying context-specific usability guidelines to interface design. Interacting with Computers. Elsevier. Vol. 12, No. 3, pp. 225-243.

[Henry, 04] Henry, S.L. and Grossnickle, M. (2004). Accessibility in the User - Centred Design Process, Georgia Tech Research Corporation, Available at http://www.uiaccess.com/accessucd/

[Hoffman, 05] Hoffman, D., Grivel, E., Battle, L. (2005). Designing software architectures to facilitate accessible Web applications. IBM Systems Journal. Vol. 44, No.3, pp. 467-483.

[Isakowitz, 95] Isakowitz, T., Stohr, E. A., Balasubramanian, P. (1995). RMM: a methodology for structured hypermedia design. Communications of the ACM, Vol. 38, No. 8. pp. 34-44.

[ISO/IEC, 99] ISO/IEC. 13407 Human-Centred Design Processes for Interactive Systems, ISO/IEC 13407:1999 (E), 1999.

[Ivory, 03] Ivory, M.Y. (2003). Automated Web Site Evaluation: Researchers' and Practitioners' Perspectives. Kluwer Academic Publishers. Dordrecht, The Netherlands.

[Jeschke, 06] Jeschke, S. And Vieritz, H. (2006). Accessibility and Model-Based Web Application Development for e-Learning Environments. In Proceedings m-ICTE 2006. Current Developments in Technology-Assisted Education, pp. 1161-1165.

[Jokela, 03] Jokela, T., Iivari, N., Matero, J., Karukka, M. (2003). The standard of user-centred design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. In Proceedings of the Latin American conference on Human-computer interaction, pp.:53 – 60.

[Kappel, 06] Kappel, G., Pröll, B., Reich, S., Retschitzegger, W. (2006). An Introduction to Web Engineering. Web Engineering The Discipline of Systematic Development of Web Applications. John Wiley & Sons. pp. 1-21.

[Lang, 03] Lang, T. (2003). Comparing website accessibility evaluation methods and learning from usability evaluation methods. Available at http://www.peakusability.com.au/pdf/website_accessibility.pdf.

[Lazar, 04] Lazar, J., Dudley-Sponaugle, A., Greenidge, K. (2004). Improving Web Accessibility: A Study of Webmaster Perceptions. Computers and Human Behavior. Vol. 20 No. 2, pp. 269-288.

[Leporini, 06] Leporini B., Paternò F., Scorcia, A. (2006). An Environment for Defining and Handling Guidelines for the Web. In K. Miesenberger et al. (Eds.). Computers Helping People with Special Needs. Lecture Notes in Computer Science 4061. Springer, pp. 176-183.

[Leporini, 06] Leporini, B., Paternò, F., Scorcia, A. (2006). Flexible tool support for accessibility evaluation. Interacting with Computers. Vol. 18, No.5, pp. 869-890.

[Maguire, 01] Maguire, M. (2001). Methods to Support Human-Centred Design. International Journal of Human-Computer Studies, Vol. 55, No. 4, pp. 587-634.

[Mariage, 04] Mariage, C. and Vanderdonckt, J. (2004). Creating Contextualised Usability Guides for Web Sites Design and Evaluation. In R. Jacob et al. (Eds). Proceedings of the 5th International Conference on Computer-Aided Design of User Interfaces, CADUI 2004, pp. 147-158.

[Mayhew, 99] Mayhew, D.J. The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design. Morgan Kaufman. 1999.

[Mendes, 06] Mendes, E., Mosley, N., Counsel, S. (2006). The need for Web Engineering: An Introduction. Web Engineering. Springer, pp. 1-27.

[Mich, 03] Mich, L., Franch, M. Gaio, L. (2003). Evaluating and Designing the Quality of Web Sites. IEEE MultiMedia. Vol. 10, No. 1, pp. 34-43.

[Montero, 03] Montero, S., Díaz, P., Aedo, I. (2003). A Framework for the Analysis and Comparison of Hypermedia Design Methods. In Proceedings of The 21st IASTED International Multi-Conference on Applied Informatics (AI 2003), M. H. Hamza (Ed.), pp. 1053-1058.

[Murugesan, 02] Murugesan, S. and Deshpande, Y. (2002). Meeting the challenges of web application development: the web engineering approach. In Proceedings of the 24th International Conference on Software Engineering, ICSE 2002. pp. 687-688.

[Murugesan, 05] Murugesan, S. and Ginige, A. (2005). Web Engineering: Introduction and Perspectives. Web Engineering: Perspectives and Techniques. Idea Group, pp. 1-30.

[Nielsen, 93] Nielsen, J. Usability Engineering. Academic Press Professional, Boston, MA. 1993.

[Nielsen, 94] Nielsen, J., Mack, R. Usability Inspection Methods. John Wiley & Sons, NY. 1994.

[Olsina, 02] Olsina, L., Rossi, G. (2002). Measuring Web Application Quality with WebQEM. IEEE Multimedia. Vol. 9, No. 4, pp. 20-29.

[Pemberton, 03] Pemberton, S. The Kiss of the Spiderbot. (2003). ACM Interactions, Vol. 10, No. 1., pp. 44.

[Petrie, 06] Petrie, H., Hamilton, F., King, N., Pavan, P. (2006). Remote usability evaluations with disabled people. In Proceedings of the SIGCHI conference on Human Factors in computing systems, CHI 2006, pp. 1133-1141.

[Plessers, 05] Plessers, P., Casteleyn, S., Yesilada, Y., De Troyer, O., Stevens, R., Harper, S., Goble, C. (2005). Accessibility: A Web Engineering Approach, In Proceedings of the 14th International World Wide Web Conference (2005), pp. 353-362.

[Rubin, 94] Rubin, J. Handbook of Usability Testing. John Wiley & Sons, NY. 1994.

[Schwabe, 95] Schwabe, D. and Rossi, G. (1995). The Object-Oriented Hypermedia Design Model. Communications of the ACM, Vol. 38, No. 8, pp. 45-46.

[Sommerville, 92] Sommerville, I. (1992). Software engineering. Addison-Wesley, Reading, MA.

[Stephanidis, 01] Stephanidis, C. (2001) The concept of Unified User Interfaces. In: C. Stephanidis (ed.) User Interfaces for All - Concepts, Methods, and Tools. Lawrence Erlbaum Associates, pp. 371-388.

[Stephanidis, 98] Stephanidis, C., Akoumianakis, D., Sfyrakis, M., & Paramythis, A. (1998). Universal accessibility in HCI: Process-oriented design guidelines and tool requirements. In C. Stephanidis & A. Waern (Eds.), Proceedings of the 4th ERCIM Workshop on User Interfaces for All.

[Sullivan, 00] Sullivan, T. and Matson, R. (2000). Barriers to use: usability and content accessibility on the Web's most popular sites. In Proceedings of the 2000 ACM Conference on Universal Usability, pp. 139-144.

[Vanderdonckt, 05] Vanderdonckt, J. and Bereikdar, A. (2005). Automated Web Evaluation by Guideline Review. Journal of Web Engineering. Rinton Press. Vol. 4, No. 2, pp. 102-117.

[Vanderdonckt, 95] Vanderdonckt, J. (1995). Accessing guidelines information with Sierra. In K. Nordby, P. H. Helmersen, D. J. Gilmore, S. A. Arnesen (Eds.). In Proceedings of IFIP TC13 International Conference on Human-Computer Interaction. INTERACT '95, pp. 311-316.

[Velleman, 06] Velleman, E., Velasco, C., Snaprud, M., Burger, D. (Eds.), Unified Web Evaluation Methodology (UWEM 1.0), 2006, Available at http://www.wabcluster.org/uwem1/UWEM_1_0.pdf.

[Vigo, 07] Vigo, M., Arrue, M., Brajnik, G., Lomuscio, R., Abascal, J. (2007). Quantitative Metrics for Measuring Web Accessibility. In Proceedings of the 2007 International Cross-Disciplinary Workshop on Web Accessibility, W4A 2007, pp. 99-107..

[Yesilada, 04] Yesilada, Y., Harper, S., Goble, C., Stevens, R. Screen Readers Cannot See (Ontology Based Semantic Annotation for Visually Impaired Web Travelers). (2004). In Proceedings of the International Conference of Web Engineering ICWE 2004, pp. 445-458.