Supporting accessible User Interfaces using Web Services

Georgios Bouloukakis*
Inria Paris-Rocquencourt, France

Ioannis Basdekis
Institute of Computer Science - Foundation of Research and Technology (ICS-FORTH), Greece

Constantine Stephanidis
Institute of Computer Science - Foundation of Research and Technology (ICS-FORTH), Greece
Department of Computer Science, University of Crete, Greece

ABSTRACT

Web services are an emerging technology which has attracted much attention from both the research and the industry sectors in recent years. The exploitation of web services as components in web applications facilitates development and supports applications interoperability, regardless of the programming language and platform used. However, existing web services development standards do not take into account the fact that the provided content and the interactive functionality should be accessible to, and easily operable by, people with disabilities. This book chapter presents a platform named myWebAccess, which provides a mechanism for the semi-automated "repair" of web services' interaction characteristics in order to support the automatic generation of interface elements that conform to the de facto standard of the Web Content Accessibility Guidelines 2.0. myWebAccess enhances interaction quality for specific target user groups, including people with visual and motor disabilities, and supports the use of web services on diverse platforms (e.g., mobile phones equipped with a browser). The web developers can build their own design templates and the users of myWebAccess can create a personalized environment containing their favourite services. Thus, they can interact with them through interfaces appropriate to their specific individual characteristics.

Keywords: Web Services, Web Accessibility, Web Engineering, User Interface, Standards

INTRODUCTION

The number of World Wide Web users is growing steadily over the last decade, and its usage is permeating all aspects of daily life. The web has a great potential to improve the quality of life of citizens, by facilitating the provision of services in basic aspects such as employment, education, marketplace and health. Especially for citizens who face barriers in accessing physical services, the web, if utilized properly, offers one – and perhaps the only – alternative solution. These citizens include:

- People with disabilities, who constitute more than 10% of the world's population. About one billion of people face problems of physical access on a daily basis[i].
- Aging people, whose number is increasing to the point where one in five people in the European Union is considered elderly[ii].
- Citizens with temporary inability to exploit some senses due to health problems.
- People with a low level of understanding and experience of technology.

* The Work reported in this book chapter was conducted when the first author was affiliated with ICS-FORTH, Greece.

The World Wide Web Consortium (W3C) has established the "Web Accessibility Initiative" (WAI-W3C), whose main objective is to provide solutions to Web Accessibility for people with visual, hearing, physical, cognitive and neurological disabilities. The results[iii] of this initiative include mainly technical guidelines such as the Web Content Accessibility Guidelines (WCAG), the Authoring Tool Accessibility Guidelines (ATAG), and the User Agent Accessibility Guidelines (UAAG). In addition, Mobile Web Best Practices provide generic instructions for building applications for mobile devices (Chuter & Yesilada, 2009). WCAG is the most renowned de facto standard, as it provides guidelines on how to create accessible interface and content elements in such a way that they can be read and manipulated by assistive technology solutions. Moreover, their applicability it facilitates interoperability with new and emerging technology solutions (e.g., navigator with voice recognition for car drivers). It is worth pointing out that some of these guidelines are included in policies all over the world (e.g., the international standard ISO/IEC 40500:2012, which is exactly the same as the WCAG 2.0, is part of an EU proposal for a directive[iv]). However, one major problem is that their application requires manual forethought, since existing development tools are not directly support compliant code (and outcome) generation.

Currently, the typical production of a web service (Alonso et al, 2003) as a function can be achieved through several development tools, greatly influencing the quality of the produced web content. In addition, interface and other content elements are integrated by the responsible development team or somehow inherited by the environment in use, so that every service acquires a similar look and feel presentation. However, a great advantage introduced by web services is the fact that their function can be reutilized and their input and output elements can be treated at other stages and by other development teams without having specific knowledge of how this web service operates, but only of the results it produces. Therefore, to support accessibility, web service core elements responsible for presentation issues must have an appropriate degree-of-freedom in order to:

- Support integrated and independent cooperation with assistive technologies (e.g., interoperability with screen readers, large keyboards, alternative pointing devices).
- Exhibit user interface which can adapt to users' preferences.
- Generate user-friendly auxiliary functionality (e.g., forms which can be filled out by using only the keyboard and quick access to areas of content).
- Upgrade with technological development (i.e., forward compatibility).
- Use and produce valid meta-language (e.g., HTML, XHTML).

Despite the proven usefulness of WCAG for web accessibility, it is common for web content manufacturers to ignore or overlook them, thus limiting the ability of disabled users to navigate to the information and services offered by a website. Thus, the aforementioned principles are far from being integrated, even to public web sites where legislation enforces them. Diachronically, studies reveal that web accessibility metrics are worsening worldwide [(Nomesa 2006), (Basdekis et al, 2010)]. Web services provide a standard form of communication between different software applications, which in turn support user's interaction through a GUI layout. They play a major role in providing content and services over the internet. The use of web services has been enhanced through standardization efforts which have resulted in interoperability specifications. However, the standards developed so far do not take into account the fact that the content and functionality that a web service offers should be accessible to people with disabilities.

For instance, the typical web service presented in Figure 1 is incorporated as a component to the www.in.gr website and provides the meteorological weather forecast for Athens. By inspecting the relevant "img" tag of the source code, it can be noted that there is no alternative description. Therefore, although it is easy for the "able-bodied" to understand the provided information, this information is not available to a blind user interacting through a screen reader.

Figure 1. Meteorological weather forecast web service: image without equivalent alternative text to inform the user about the cloudy evening.

Aiming to fill in such accessibility gap, this paper contributes practically applicable solutions to ensure that web services have all the presentation characteristics required to render them accessible to people with disabilities. In the context of this work, by analyzing the standards for web services, the description of data was examined, in order to implement rules that indicate the “additional metadata”. The final service should be enriched with such “additional metadata” in order to comply with accessibility guidelines when incorporating it into a web site.

The findings from the above analysis provided the specifications for the design and implementation of a system that is able to semi-automatically repair problematic web services, and offer them through a web application in a uniform and user-friendly manner compatible with assistive technology. In addition, this system automatically adapts content generated by third parties and provides it to various browser-equipped devices, using different personalization options for each end-user.

BACKGROUND

It is known that basically a web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with a web service in a manner prescribed by its description using SOAP (Simple Object Access Protocol) or REST (REpresentational State Transfer) messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards.

Web services support application interoperability regardless of platform and programming language (Papazoglou, 2008). They use standards based on the XML language in order to describe the exchanged data with other applications, through the operation that has been invoked. Web services may offer a wide variety of information (e.g., theatre review articles, weather reports, credit checks, etc) in many forms (e.g., text, image, data tables, interactive forms, etc) to support a variety of devices (e.g., desktop, mobile, iTV, etc). In the future, some of the most interesting XML web services will - and maybe force to - support solutions for applications in order to overcome interoperability challenges of heterogeneous information systems. To support this argument, for example, healthcare applications are already copying with the problem of adaptation and interaction with existing systems in a constantly changing health environment because of the endless introduction of new devices. The enriched web services for those applications [(Barbarito et al, 2012), (Mykkanen et al, 2005)] will offer seamless and high quality care to the patients through a web application, especially to people with disabilities.

A study regarding the Return of Investment (ROI) of delivering e-accessibility (Brinck, 2008), made clear that achieving accessible designs should become more affordable over time, as awareness increases and more designer/developers receive appropriate training. However and despite this evidence, the majority of software vendors still seem reluctant to incorporate or have low awareness of e-accessibility benefits, and almost no awareness at all of the possible approaches that they can follow to achieve it. Moreover, even the “convinced” web developers who find themselves in the challenging position to (re)-produce a web service following accessibility requirements, have limited knowledge about alternative approaches and issues that each solution involves in order to enhance the development processes (Basdekis et al., 2005). Unfortunately, methodologies, guidelines and supporting tools have - till now -failed to give practical guidance on supporting e-accessibility to inexperienced web service developers and providers. Inevitably, the notion of e-accessibility is generally undervalued in the web services domain, however sporadic efforts by the research community provide solutions to address this issue. Examples include an existing personalization system which contains as components the desired services of a user, as well as research efforts towards automatic content adaptation depending on the access device and the profile of the user. In particular, these efforts are:

A. Accessibility in the Web Services domain. To achieve the exploitation of web services by all users, including people with disabilities, (Giakoumis et al, 2011) proposed a tool for assessing the accessibility degree of a web service. By using this tool, it is possible to assess whether a SOAP or REST web service conforms to the WCAG. In particular, based on the presented functionality and due to the limited requirements of the WSDL (Web Service Definition Language) standard coupled with the common practice of developers that does not fully exploit the specifications during development, a SOAP type web service will rarely be assessed as fully accessible. Also, in the case of an input REST type web service, and due to the absence of a standard for this type (REST), the tool provides an interface through which a user is able to create a specification called WADL. The problem of this procedure is that the user puts significant effort with the risk not to define the necessary interaction metadata based on WCAG. Finally, the above tool provides only information about the accessibility degree of a web service, without proposing a solution to repair it. Furthermore, a system architecture based on services has been developed, which offers easier navigation in a city for motor impaired users. Based on the functionality of this system, it was noticed that the specifications which define the format of data in a WSDL are quite limited, especially in the case of annotating them with semantic information.

B. Personalized Interfaces. An optimal environment for all internet users would be a web application that will collect all third party services which conform to WCAG 2.0 (Caldwell et al, 2008) in order to interact with assistive technology solutions. Additionally, the possibility of customizing the system for certain categories of users, including people with vision or motor impairment, would provide the necessary degree of personalization to achieve a friendly interface. There are many popular web applications that provide personalized interfaces with the favourite gadgets of each user as components (i.e., Netvibes, iGoogle, MyYahoo, WebWag and Gritwire). None of those applications uses third party services for their functionality, except some commercial services such as iGoogle, without conforming their interface to WCAG.

C. Automatic adaptation of content with proxy as system. Besides offering online applications that collect third party services, there are systems that use various methods to analyze a website in order to convert the extracted content to another format. These applications usually operate as a proxy by making an analysis of the web mark-up. Then, they reclassify the content while making some sort of corrections to continuously improve the accessibility. Examples of such systems are:

- The BBC service named Betsie. By using this service, the BBC website content became easier to read for blind and color-blind users (Brown & Robinson, 2001).
- The WebFACE tool (Alexandraki et al, 2004), through which extra features are added to enhance the accessibility. However, it is applied only to specific web page structures as the one of (Maeda et al, 2004).
- A system for dynamically updating web pages to achieve a high contrast background-foreground, delete images and parts of the context that do not conform with WCAG 2.0 and offer an alternative navigation (Richards & Hanson, 2004).

All mentioned systems, by using some form of heuristic algorithms, parse the content and then adjust and rearrange it by using appropriate tags. The success of the final result depends on how the initial version of the website conforms to valid XHTML meta-language. In an effort to provide a different methodological approach, this book chapter is an extended version of the short paper (Bouloukakis et al, 2013), and introduces a platform named *myWebAccess*. The method involves an improved way to describe the metadata transferred via a web service, a repair mechanism to support e-accessibility requirements and finally the automatic generation of the appropriate web form for the delivery of the service.

MYWEBACCESS PLATFORM

There are several ways to describe the data transferred via a web service, such as the DTD[v], XML Schema[vi], RSS[vii] and WSDL[viii]. WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described at an abstract level, and subsequently are bounded to a concrete network protocol where each message has a specific structure for an endpoint (Papazoglou, 2008). In other words, the WSDL standard corresponds to a web service that specifies exactly the input/output and the necessary procedure to invoke it. The work reported in this book chapter utilizes the WSDL standard in order to improve content accessibility.

However, even in the case of using the WSDL specification, there are technical limitations. More specifically, developers in most cases fail to provide all the necessary information, since there are cases where WSDL does not cover exhaustively all situations, and as a result presented metadata are limited in order to conform to the specific provisions of the WCAG. An example is a web service which creates an image displaying the text given as input by the user. By using the WSDL specification, the input of the web services is represented by some parameters such as height (Integer), width (Integer), color (String) and text (String). In order to fill the above fields the WSDL standard does not completely determine the unit of measurement for the height of the image (pixel or inches). Also, there is no restriction on the font size of the image text. Finally, the color model (e.g., RGB, HSL, CMYK) that should be used is not clear. In order to support the provision of "correctly constructed" web services in practice, and make their usage easy and convenient by people with disabilities, the following should be provided:

- Possibility of adding of web services (add).
- Possibility of enriching parameters dealing with presentation or interaction elements (repair).
- Generation of the input and the outcome of the web service in a way that conforms to WCAG (adapt).

Thus, it seems necessary to create an information system able to scan the available functionality of a web service and include the related components. This system should provide a mechanism to add metadata in order to augment web services with accessibility features. Additionally, the preferences stored in the user profile will further facilitate the interaction of end-users with the services and will provide easier access per user category (e.g., profiles for people with dyschromatopsia). At the same time, and in accordance with recent technological advances, any web service should be able to adapt to user device's capabilities.

In the context of this work, a platform called *myWebAccess* is proposed and implemented. The platform provides a mechanism for adding and semi-automatically "repairing" third-party web services, so that all the interaction elements can be determined and thus content generation complaint to WCAG 2.0 can be achieved.

Towards ensuring accessibility characteristics in Web Services

Web services can be classified in two categories according to their interaction with the user. The first category includes web services that present information (**outcome**) without any user input, while the second requires implicitly or explicitly some **input** prior to the presentation of the outcome. For example, a weather forecast is a service of the second category that requires the name of the desired city through text input. Taking into account the interaction behavior of a user utilizing assistive technology, the input/outcome of such service should provide several technical features that enhance accessibility. More specifically, and besides adhering to WCAG 2.0:

- A convenient navigation mechanism need to be provided (e.g., extra features for motor impaired users).
- Balanced color contrast between foreground and background should be used increase text readability, with potential fluctuation of the font size.
- In the case of non-text context, an equivalent alternative text description should be provided for images, graphics and multimedia.
- Data tables must involve caption and summary tags.
- In the case of interactive forms, each "input" tag should have labels to be properly announced and navigational aids should speed up navigation between fields. In addition, a mechanism for validating user input should be provided for error prevention.
- Additional adaptation features can be triggered depending on the device used (e.g., width of a mobile screen).

Summarizing, the input/outcome of a web service should be suitably adapted to the specific preferences of the user and the technical capabilities of any screen in use. To improve the integration of a web service in an application, appropriate metadata characterizing the input/outcome elements should be present.
This additional information (**metadata**) should be provided somehow (e.g., by the manufacturer or at a later stage by a service administrator) in order to overcome limitations. In order to create a web service which will contain all the necessary interaction information to appropriately interact with the user, the implementation of the following steps was deemed necessary: (i) analysis of the WSDL file, (ii) separation of the parameters that needed the additional metadata, (iii) import of metadata for each parameter based on the WCAG and (iv) creation of an XML file that defines the additional metadata.
As depicted in Figure 2, the generated XML is directly linked to the originated WSDL, and both files will provide the necessary information for the exploitation of accessible web services. To evaluate the effect of the above procedure, the ideal scenario would be to analyze a web service and request suggestions from users experienced in web accessibility issues (Reich, 2009). These suggestions would indicate the additional metadata needed. Thus, an experienced web service administrator is needed to repair web services that contain poor or insufficient interaction metadata.

Figure 2. Procedure to add accessibility features.

Platform requirements to import Web Services

One of the basic components of the platform is web services management and, more specifically, operations management, where operations can be added, edited and become available to the end-users. As already mentioned, in a web service, the information that is needed to perform an operation is defined in the WSDL file, which is an XML Schema (XSD) (Papazoglou, 2008) published on the web using a URL.
According to the steps described in Figure 2, the process followed for semi-automated repair of a web service and its functions is the following. The WSDL file is given as an input to a parser (step ① in Figure 3) that defines a variety of meta-data for each data type (i.e., input/output parameter) and correspond to a web service. Next, the additional metadata are analyzed[ix] (step ② in Figure 3) and presented to the system administrator by using a web interface that indicates the "gaps" that should be filled for each input/output parameter (step ③ in Figure 3). Next, the system generated or manually inserted metadata are stored in an XML file (step ④ in Figure 3). The WSDL and the additional metadata (XML file) are directly connected to each other, so that presentation elements conforming to the WCAG can be generated (step ⑤ in Figure 3). Finally, the end users are able to interact with services, since they are integrated with valid and compliant mark-up (step ⑥ in Figure 3). This process ensures the proper enrichment of the parameters prior to the publication and availability of the web service and its functions.

Figure 3. Process to import a web service to the myWebAccess platform.

The exploitation of accessible web services as components in a web application has the following advantages:

- **Reuse**: reduce manufacturing costs and availability [(Basdekis et al, 2005), (Clark, 2003)].
- **Multi-device**: easier adaptation of the presented information to various devices [(Basdekis et al, 2009), (Karampelas et al, 2009), and (Chiti & Leporini, 2012)].
- **Conformity with standards**: achievement of interoperability with assistive technology solutions.
- **Cost**: lower upgrade costs accordance to the technological development (Sierkowski, 2002).

SUPPORT OF ACCESSIBLE AND MULTI-CHANNEL WEB INTERFACE GENERATION

The majority of web applications adopt a specific content structure during construction (Curtis, 2009). More specifically, according to common practice followed in recent years, there are four main areas (i.e., Header, Sidebar, Main content and Footer as shows the design template ① in Figure 4). Those areas usually have the same look & feel on all pages of a website. Furthermore, this structure can easily reutilize different design templates. By setting specific content areas in advance, the initial structure (design template ① in Figure 4) can be enriched with additional navigation aids (design template ② in Figure 4), or in the case of a display on a mobile device, the initial structure can split into more pages for improved readability (design template ③ in Figure 4). In both cases, the developer has to ensure the proper adaptation of the main content that is updated frequently.

Figure 4.Various design templates for websites.

As mentioned earlier, web services can be treated in a web application as components where functionality is provided by third parties (Figure 5), resulting in easier adaptation to different design templates. Thus, appropriate mechanisms are necessary to present web services for different third parties in a unified presentation schema.

Figure 5. Web services as components to a website.

Platform Architecture

The purpose of most information systems is to retrieve data from a source and display it to the user. If the “user interface” and the “logic application” are connected to a single part/object of the application, then whenever the user requires a change in the user interface, the whole section containing the “logic” is modified too (Fielding, 2000).

The Model-View-Controller (MVC) architecture (Reenskaug, 2003) addresses this issue by separating the display of the data, the functionality of the application and the data storage. The *myWebAccess* platform has been built using the MVC architecture to support multiple design templates. More specifically, as depicted in Figure 6, the View is responsible for the display of data and the Model for storing this data. Finally, the Controller undertakes the handling of a user request, namely, the data recovery and the selection of the appropriate design template.

For instance, suppose that a user wants to access the platform through his mobile phone to get the latest news from Yahoo. The controller analyzes the request and detects the access device, retrieves the data and updates the user interface by selecting the appropriate design template. The same procedure is followed for a request from a tablet device, but a template for larger touch-screens is selected (Figure 6).

Figure 6. myWebAccess architecture

Adaptation of services at different context of use

After the insertion of a web service by the system administrator, the *myWebAccess* platform provides automatically generated content (step ⑤ in Figure 3) that is compliant to the WCAG 2.0 technical standards. This content is enriched with semantic information (if required), is adaptable to support specific interactions, and has the ability to invoke this service. Thus, taking advantage of the above, *myWebAccess* creates a suitable interface for each user category by serving:

1. **Blind users** with content that conforms to the WCAG 2.0, level AA.
2. **Users with impaired vision** or **color blindness** with high contrast background – foreground.

3. **Motor impaired users** with
   (a) sliding navigation (Adams et al, 2007), (Myers et al, 2002).
   (b) sliding navigation with a virtual keyboard (Norte & Lobo, 2007), (Zhai et al, 2000).
4. Supporting **mobile devices**.

The architecture of the *myWebAccess* platform provides the necessary degree of freedom to address the diversity of user requirements and devices, covering the different scenarios derived from the combination of these two factors. This is achieved through a library of alternative design templates and the classification of users into different profiles. For instance, in order to facilitate motor impaired users, the library contains a suitable design template and provides sliding navigation and a virtual keyboard [(Adams et al, 2007), (Norte & Lobo, 2007)], (Figure 7).

Figure 7.Sliding navigation design template with a virtual keyboard

This enables the user to navigate the web page by using only one button. For example, in order to fill in the field named "Number 1", the sliding window contains the corresponding hyperlink and through it the virtual keyboard is displayed. Then, by using only two buttons (binary switches (Norte & Lobo, 2007)) the user can fill in the field. It is worth noting that without the repairing of services by the *myWebAccess* platform (Figure 3), the mark-up of the operation "add" would not be valid, and therefore it could not be invoked by using only two buttons.

Use by the web developer

Targeting facilitated extensibility of our solution, we provide a highly-optimized architecture, where each web service constitutes the common reusable part, giving the ability to the web developer for introducing a new design template. Therefore, a developer wishing to add a new service on the *myWebAccess* platform should play the system administrator's role in order to use the mechanism for adding third-party web services. Then, the already "repaired" services can be utilized as components to the existing design templates. There is the possibility to create the design template by following the specific content structure. In the *myWebAccess* platform, we build a specific content structure named "default-design-template". Thus, each developer is able to create the specific main areas (header, sidebar, main content and footer) with the desired presentation options that will be compliant to the WCAG 2.0 technical standards (due to the default design template).

EVALUATION

This section describes the process and the results of the accessibility and usability assessment of the *myWebAccess* platform. Specifically, it is examined whether the proposed technical solution produces the desired results in accordance with the standards.

Evaluation Methodology

To assess the level of compliance of the services available to the end-user to WCAG 2.0, *myWebAccess* was tested using semi-automated accessibility testing tools. Three categories of web services were tested: (i) news feeds via RSS, (ii) meteorological weather forecast, and (iii) image generator. More specifically, proper display was checked on different browsers and devices, the usage of meta-language content was validated using the W3C validation tool[x] and compliance with WCAG 2.0 was evaluated using the Web Accessibility Test (T.A.W.) tool[xi]. Regarding the validation of the used meta-language (XHTML), the generated web pages are constructed according to the W3C standards of XHTML and CSS. All the generated content complies with the accessibility guidelines WCAG 2.0 for all web pages of the *myWebAccess* platform. The results show that there were no problems observed, except for 30 notes.

These notes cannot be controlled by the tool and require the manual inspection of a specialist. For instance, a specialist has to check the description of the images and the page title that should be descriptive. Furthermore, the headers on each page, the labels and the forms should be well structured and quite descriptive. The combination of the selected colours and the degree of the contrast as well as the existence of alternative navigation on the website are all tested and were found to comply.

After making any adjustments arising from the results of the above-mentioned tests, the second step consisted of evaluating the usability with end users. In this context, a scenario of use was elaborated for evaluating the proper service performance, as well as the usability and usefulness of the service for each user category. This scenario contains three tasks for the selection, management and invocation of "repaired" services in the *myWebAccess* platform. Ten experts in the use of assistive technology were involved using profiles visual or motor impaired users. The scenario was executed in our laboratory using one computer to switch the profiles depending on the user.

All users completed the scenario within an acceptable period of time. The maximum time was 22 minutes. As depicted in the diagram of Figure 8, those using a screen reader were slower compared to the rest of the subjects and carried out tasks in two to four timeframes. This difference in performance was expected, since in this case the user is forced to listen to an important part of the content before understanding at which point that part is, while filling in data forms requires a special procedure. However, on the whole, everyone finished their task within reasonable timeframes.

The number of errors made by each user was also measured. Figure 9 illustrates in detail the number of errors that experienced users made throughout the process. As expected, a high number of errors were observed in the category of users who use a screen reader. Especially in the case of the scenario simulating the behavior of a user with upper limb disability by using binary switches, a remarkable difference was noticed in the execution time compared to the rest of the users. The tasks performed by two users with prior experience in the use of these devices (binary switches). In this use case, an additional navigation with fast access hyperlinks (Sliding Navigation) was activated (Figure 7).

Figure 8. Required time (min) for the completion of task.

Figure 9. Number of errors for each user.

Figure 10 shows that, for each task usage scenario, the results of the execution time, between the user 1 and user 2, are similar. Compared to the rest of the users, the time is almost four times greater. This difference occurs due to the fact that the users were asked to fill data in forms using the virtual keyboard (with sequential selection of characters).

Figure 10. Time (min) for completing three tasks.

Summarizing, it appears that users who use assistive technology are able to achieve sufficient interoperability with the offered web services.

FUTURE RESEARCH DIRECTIONS

Following an extensive user-based evaluation the following extensions could be incorporated into the system in the future:

- Exporting to a script with XHTML valid mark-up by taking the metadata from the existing WSDL and XML files which have been created by the *myWebAccess* platform.
- Further exploitation of the Semantic Web (Berners-Lee et al, 2001), and Semantic Web Services (McIlraith et al, 2001) towards a more comprehensive solution that means transition from the WSDL standard, to a standard with semantic information (OWL-S[xii]).
- The ability to exploit REST type web services.
- The creation of design templates for more use case environments covering specific disability, and modern devices (e.g. digital TV).
- The possibility to verify automatically WCAG's check point (concerning also the manual inspection of a specialist).
- Extend the platform to automatically repair one or more similar operations.
- Including the Accessible Rich Internet Applications Suite (WAI ARIA[xiii]) to improve the interaction via assistive technology (faster and easier).
- Evaluating the web services by using the "quality of service" and "quality of experience" in web services domain [(O'Sullivan et al, 2002), (Giakoumis et al, 2013)].
- Creating different registries of "repaired" services by leveraging QoS aspects (O'Sullivan et al, 2002) to support varied disability requirements.

CONCLUSION

The main idea behind the work presented in this book chapter is the exploitation of web services as components in a web interface in order to facilitate the development of accessible and multi-channel web interfaces. A technical process has been elaborated in order to enable accessibility characteristics in the web services domain. Leveraging on this process, the *myWebAccess* platform for repairing, enhancing and redistributing web services accessible to people with disability has been developed. Moreover, the use of the platform has illustrated the adaptation of the repaired services in different context of use. Finally, the proposed approach has been evaluated by examining web services reusability and interoperability with assistive technology solutions.

REFERENCES

Adams, C., Collison, S., Budd, A., Boulton, M., & Clarke, A. (2007). *Web standards creativity: innovations in web design with XHTML, CSS, and DOM scripting* (Vol. 24). friends of ED.

Alexandraki, C., Paramythis, A., Maou, N., & Stephanidis, C. (2004). Web accessibility through adaptation. *Computers Helping People with Special Needs*, 626-626.

Alonso, G., Casati, F., Kuno, H., & Machiraju, V. (2003). *Web services: concepts, architectures and applications.* Springer.

Barbarito, F., Pinciroli, F., Mason, J., Marceglia, S., Mazzola, L., & Bonacina, S. (2012). Implementing standards for the interoperability among healthcare providers in the public regionalized Healthcare Information System of the Lombardy Region. *Journal of Biomedical Informatics*.

Basdekis, I., Alexandraki, C., Mourouzis, A., & Stephanidis, C. (2005). Incorporating Accessibility in Web-Based Work Environments: Two Alternative Approaches and Issues Involved. In C. Stephanidis, (Ed.), Universal Access in HCI: Exploring New Dimensions of Diversity - Volume 8 of the Proceedings of the 11th International Conference on Human-Computer Interaction (HCI International 2005), Las Vegas, Nevada, USA, 22-27 July. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Basdekis, I., Karampelas, P., Doulgeraki, V., and Stephanidis, C. (2009). Designing Universally Accessible Networking Services for a Mobile Personal Assistant. In C. Stephanidis, (Ed.), Universal Access in Human-Computer Interaction - Intelligent and Ubiquitous Interaction Environments. – Volume 6 of the Proceedings of the 13th International Conference on Human-Computer Interaction (HCI International 2009), San Diego, CA, USA, 19-24 July, pp. 279-288. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 5615, ISBN: 978-3-642-02709-3).

Basdekis, I., Klironomos, I., Metaxas, I., & Stephanidis, C. (2010). An overview of web accessibility in Greece: a comparative study 2004--2008. *Universal Access in the Information Society, 9*(2), 185-190.

Berners-Lee, T., Hendler, J., Lassila, O., & others. (2001). The semantic web. *Scientific american, 284*(5), 28-37.

Bouloukakis, G., Basdekis, I., Stephanidis, C. (2013). myWebAccess: A platform for repairing, enhancing and re-distributing Web Services accessible to people with disability. *IADIS-International Conference on Interfaces and Human Computer Interaction.*

Brinck, T. (2008). Return on Goodwill: Return on Investment for Accessibility. In S. Harper, and Y. Yesilada (Eds.), *Web Accessibility – A Foundation for Research* (pp. 385-414). Berlin Heidelberg: Springer.

Brown, S., & Robinson, P. (2001). A world wide web mediator for users with low vision. *CHI'2001 Conference on Human Factors in Computing Systems Workshop.*

Caldwell, B., Cooper, M., Reid, L., & Vanderheiden, G. (2008). Web content accessibility guidelines 2.0. *W3C Recommendation, 11*.

Chiti, S., & Leporini,B. (2012). Accessibility of android-based mobile devices: a prototype to investigate interaction with blind users, Proceedings of the 13th international conference on Computers Helping People with Special Needs, July 11-13, 2012, Linz, Austria

Chuter, A., & Yesilada, Y. (2009). Relationship Between Mobile Web Best Practices (MWBP) and Web Content Accessibility Guidelines (WCAG).

Clark, J. (2003). *Building accessible websites Indianapolis: New Riders.* Tech. rep., ISBN 0-7357-1150-x.

Curtis, N. (2009). *Modular Web Design: Creating Reusable Components for User Experience Design.* New Riders Pub.

Fielding, R. (2000). *Architectural styles and the design of network-based software architectures.* Ph.D. dissertation, University of California.

Giakoumis, D., Votis, K., & Tzovaras, D. (2013). Introducing web service accessibility assessment techniques through a unified quality of service context. *Service Oriented Computing and Applications*, 1-16.

Karampelas, P., Basdekis, I., and Stephanidis, C. (2009). Web User Interface Design Strategy: Designing for Device Independence. In C. Stephanidis, (Ed.), Universal Access in Human-Computer Interaction - Addressing Diversity – Volume 5 of the Proceedings of the 13th International Conference on Human-Computer Interaction (HCI International 2009), San Diego, CA, USA, 19-24 July, pp. 515-524. Berlin Heidelberg: Lecture Notes in Computer Science Series of Springer (LNCS 5614, ISBN: 978-3-642-02706-2).

Maeda, J., Fukuda, K., Takagi, H., & Asakawa, C. (2004). Web accessibility technology at the IBM Tokyo Research Laboratory. *IBM journal of research and development, 48*(5.6), 735-749.

McIlraith, S., Son, T., & Zeng, H. (2001). Semantic web services. *Intelligent Systems, IEEE, 16*(2), 46-53.

Myers, B. A., Wobbrock, J. O., Yang, S., Yeung, B., Nichols, J., & Miller, R. (2002). Using handhelds to help people with motor impairments. *Proceedings of the fifth international ACM conference on Assistive technologies* (pp. 89-96). New York, NY, USA: ACM.

Mykkanen, J., Riekkinen, A., Laitinen, P., Karhunen, H., & Sormunen, M. (2005). Designing web services in health information systems: From process to application level. *Studies in health technology and informatics, 116*, 515.

Nomensa: United Nations global audit of web accessibility (2006). [On-line]. Available at: http://www.un.org/esa/socdev/enable/documents/fnomensarep.pdf [Accessed 12 Feb 2014].

Norte, S., & Lobo, F. (2007). A virtual logo keyboard for people with motor disabilities. *ACM SIGCSE Bulletin, 39*, pp. 111-115.

O'Sullivan, J., Edmond, D., & Ter Hofstede, A. (2002). What's in a Service? *Distributed and Parallel Databases, 12*(2), 117-133.

Papazoglou, M. (2008). *Web services: principles and technology.* Addison-Wesley.

Reenskaug, T. (2003). The Model-View-Controller (MVC) Its Past and Present Trygve Reenskaug, University of Oslo.

Reich, K. (2009). eInclusion, eAccessibility and design for all issues in the context of European Computer-Based Assessment. *The Transition to Computer-Based Assessment*, 68.

Richards, J., & Hanson, V. (2004). Web accessibility: a broader view. *Proceedings of the 13th international conference on World Wide Web*, (pp. 72-79).

Sierkowski, B. (2002). Achieving web accessibility. *Proceedings of the 30th annual ACM SIGUCCS conference on User services*, (pp. 288-291).

Zhai, S., Hunter, M., & Smith, B. A. (2000). The metropolis keyboard - an exploration of quantitative techniques for virtual keyboard design. *Proceedings of the 13th annual ACM symposium on User interface software and technology* (pp. 119-128). New York, NY, USA: ACM.

ADDITIONAL READING

Abou-Zahra, S. (2012). WAI-ACT: web accessibility now. *Proceedings of the 21st international conference companion on World Wide Web*, (pp. 215-218).

Abou-Zahra, S., & Henry, S. L. (2010). Exploring web accessibility solutions in developing regions as innovations for the benefit of all. *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, (p. 5).

Aghdam, P. A., & Rafeh, R. (2013). Accessing Canvas Element in HTML5. *AWERProcedia Information Technology and Computer Science*, *1*.

Atkinson, M. T., Bell, M. J., & Machin, C. H. (2012). Towards ubiquitous accessibility: capability-based profiles and adaptations, delivered via the semantic web. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 14).

Bakalov, F., Meurs, M.-J., K{\"o}nig-Ries, B., Sateli, B., Witte, R., Butler, G., et al. (2012). Empowering web portal users with personalized text mining services. *EMBnet. journal*, *18*(B), pp--81.

Benslimane, D., Dustdar, S., & Sheth, A. (2008). Services mashups: The new generation of web applications. *Internet Computing, IEEE*, *12*(5), 13-15.

Berger, S., McFaddin, S., Narayanaswami, C., & Raghunath, M. (2003). Web services on mobile devices-implementation and experience. *Mobile Computing Systems and Applications, 2003. Proceedings. Fifth IEEE Workshop on*, (pp. 100-109).

Castanon, J., Maia, M., & Silva, M. (2012). The ambient intelligence for the sake of accessibility in residential projects: a proper study to the Brazilian scene. *Work: A Journal of Prevention, Assessment and Rehabilitation*, *41*, 5686-5688.

Cavallaro, L., & Di Nitto, E. (2008). An approach to adapt service requests to actual service interfaces. *Proceedings of the 2008 international workshop on Software engineering for adaptive and self-managing systems*, (pp. 129-136).

Cooper, M., Sloan, D., Kelly, B., & Lewthwaite, S. (2012). A challenge to web accessibility metrics and guidelines: putting people and processes first. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 20).

de Santana, V. F., de Oliveira, R., Almeida, L. D., & Baranauskas, M. C. (2012). Web accessibility and people with dyslexia: a survey on techniques and guidelines. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 35).

Fogli, D., Provenza, L. P., & Bernareggi, C. (n.d.). A universal design resource for rich Internet applications based on design patterns. *Universal Access in the Information Society*, 1-22.

Hoehl, J. (2012). Leveraging crowds and clouds: towards an intercommunity approach to improving web accessibility. *ACM SIGACCESS Accessibility and Computing*(102), 17-20.

Lai, P. P. (2011). Application of content adaptation in web accessibility for the blind. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 6).

Lazar, J., Olalere, A., & Wentz, B. (2012). Investigating the accessibility and usability of job application Web sites for blind users. *Journal of Usability Studies*, *7*(2), 68-87.

Lunn, D., & Harper, S. (2011). Improving the accessibility of dynamic web content for older users. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 16).

Minifie, D., & Coady, Y. (2009). Getting mobile with mobile devices: Using the web to improve transit accessibility. *Proceedings of the 2009 International Cross-Disciplinary Conference on Web Accessibililty (W4A)*, (pp. 123-126).

Mirri, S., Salomoni, P., & Prandi, C. (2011). Augment browsing and standard profiling for enhancing web accessibility. *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, (p. 5).

Paisios, N. (2012). *Mobile Accessibility Tools for the Visually Impaired.* Ph.D. dissertation, New York University.

Phillips, D., & Thompson, R. (2013). Limitations \& Advances in Mobile Web Application Development. *Enquiry-The ACES Journal of Undergraduate Research, 3*(1).

Rello, L. (2012). DysWebxia: a model to improve accessibility of the textual web for dyslexic users. *ACM SIGACCESS Accessibility and Computing*(102), 41-44.

Rello, L., & Baeza-Yates, R. (2012). Lexical quality as a measure for textual web accessibility. In *Computers Helping People with Special Needs* (pp. 404-408). Springer.

Salomoni, P., Mirri, S., Ferretti, S., & Roccetti, M. (2008). A multimedia broker to support accessible and mobile learning through learning objects adaptation. *ACM Transactions on Internet Technology (TOIT), 8*(2), 4.

Sathapornvajana, S., & Papasratorn, B. (2013). Effect of Culture, Age, and Language on Quality of Services and Adoption of IP Applications. *Wireless Personal Communications*, 1-9.

Sohaib, O., Hussain, W., ISMAILI, I., & BUKHARI, A. (2012). Exploring the Relationship between Web Usability and the Web Accessibility Guidelines. *Exploring the Relationship between Web Usability and the Web Accessibility Guidelines*. unpublished.

Sun, L. (2012). Web services for human interaction.

Trewin, S., Bellamy, R., Thomas, J., Brezin, J., Richards, J., Swart, C., et al. (2010). Designing for auditory web access: accessibility and cellphone users. *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, (p. 4).

Xu, H. (2013). *A Support System for Graphics for Visually Impaired People.* Ph.D. dissertation, The University of Western Ontario.

Zakraoui, J., & Zagler, W. (2012). A method for generating CSS to improve web accessibility for old users. In *Computers Helping People with Special Needs* (pp. 329-336). Springer.

---

[i] http://www.disabled-world.com/disability/statistics
[ii] http://epp.eurostat.ec.europa.eu/portal/page/portal/eurostat/home
[iii] http://www.w3.org/standards/techs/accessibility#w3c_all
[iv] http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-%2F%2FEP%2F%2FNONSGML%2BCOMPARL%2BPE-513.011%2B01%2BDOC%2BPDF%2BV0%2F%2FEN
[v] http://www.w3.org/TR/xhtml1/dtds.html
[vi] http://www.w3.org/2003/09/xmlap/xml-schema-wg-charter.html
[vii] http://en.wikipedia.org/wiki/RSS
[viii] http://www.w3.org/TR/wsdl
[ix] http://www.urdalen.no/wsdl2php/index.php
[x] http://validator.w3.org
[xi] http://www.tawdis.net
[xii] http://www.w3.org/Submission/OWL-S
[xiii] http://www.w3.org/WAI/intro/aria