Web Engineered Applications for Evolving Organizations:

Emerging Knowledge

Ghazi I. Alkhatib
Applied Science University, Jordan

Information Science REFERENCE

Senior Editorial Director:	Kristin Klinger
Director of Book Publications:	Julia Mosemann
Editorial Director:	Lindsay Johnston
Acquisitions Editor:	Erika Carter
Development Editor:	Mike Killian
Production Coordinator:	Jamie Snavely
Typesetters:	Keith Glazewski, Natalie Pronio and Milan Vracarich, Jr.
Cover Design:	Nick Newcomer

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

Library of Congress Cataloging-in-Publication Data

Web engineered applications for evolving organizations : emerging knowledge / Ghazi I. Alkhatib, editor.
p. cm.
Includes bibliographical references and index.
Summary: “This book explores integrated approaches to IT and Web engineering, offering solutions and best practices for knowledge exchange within organizations”--Provided by publisher.
ISBN 978-1-60960-523-0 (hardcover) -- ISBN 978-1-60960-524-7 (ebook) 1. Web site development. 2. Organizational change. I. Alkhatib, Ghazi, 1947-
TK5105.888.W37243 2011
006.7068--dc22
2011009995

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

Chapter 19
VoiceWeb:
Spoken Dialogue Interfaces and Usability

Dimitris Spiliotopoulos
University of Athens, Greece

Georgios Kouroupetroglou
University of Athens, Greece

Pepi Stavropoulou
University of Athens, Greece

ABSTRACT

This chapter presents the state-of-the-art in usability issues and methodologies for VoiceWeb interfaces. It undertakes a theoretical perspective to the usability methodology and provides a framework description for creating and testing usable content and applications for conversational interfaces. The methodologies and their uses are discussed as well as certain technical issues that are of specific importance for each type of system. Moreover, it discusses the hands-on approaches for applying usability methodologies in a spoken dialogue web application environment, including methodological and design issues, resource management, implementation using existing technologies for usability evaluation in several stages of the design and deployment. Finally, the challenging usability issues and parameters of the emerging advanced speech-enabled web interfaces are presented.

INTRODUCTION

Research in human-computer interaction aims at gaining an in depth understanding of the nature and principles governing the interactive communication between humans and machines, so that this understanding may be utilized in the development of universally usable and useful interfaces that address and adapt to user rather than system needs. In this line of thought enabling the use of various modalities like speech, gestures, haptics and graphical displays as input and output to such systems should enhance naturalness and ease of use.

At the same time, advances in web technologies over the past years have significantly increased the range of practical applications suited for such multimodal interaction. With high speed inter-

DOI: 10.4018/978-1-60960-523-0.ch019

net availability providing access to demanding multimodal services to all homes, a lot of people can now benefit from real-time services ranging from voice banking to online socialising and e-commerce. While most high-level services are provided solely through web pages and the traditional mouse and keyboard interface, there are, nevertheless, providers who have begun deploying spoken dialogue interfaces to new or existing web applications, acknowledging the fact that spoken dialogue is now widely considered to comprise a significant aspect of multimodal human-machine interaction and a means to increased customer satisfaction and naturalness of information access.

As with all human-computer interfaces, spoken dialogue interfaces are built with the target user in mind. Thorough requirements analysis and efficient design methodology are imperative in their case as well, especially if one takes into account the capabilities and limitations of current speech understanding technologies that should be compensated for in order to reach industrial standards. Not all technologies involved in the development process are of the same maturity and/or standardisation, and there is only a limited number of platforms available for building such systems. Thus, given the range, variability and complexity of the actual business cases it is obvious that the enabling technologies may produce working systems of variable usefulness due to design and/or implementation limitations. In addition, the use of a transient medium such as speech as the main input and output mode substantially differentiates spoken dialogue interfaces from traditional graphical user interfaces (GUIs) and web interfaces. Therefore, even though core usability principles may in general apply, there are particular to the development of speech based web interfaces considerations, principles, guidelines and techniques that simply render the direct translation of a non-speech user interface into a speech-based interface infelicitous. Indeed spoken dialogue far more enhances naturalness in comparison to using forms and buttons on a traditional web interface. However, is the user satisfaction similarly improved? Does the performance of the resulting application meet the user requirements? How is usability ensured by design and verified by evaluation in a spoken dialogue web interface?

This chapter discusses the background of speech-based human-computer interaction and elaborates on the spoken dialog interfaces and the ways they differ from traditional web interfaces. It explores what usability is and how it is ensured for natural spoken dialogue interaction interface design and implementation. Finally, it presents key methodologies for usability testing of spoken dialogue web interfaces and discusses some of the challenges posed by the use of speech as the main modality in light of a speech-enabled complex application.

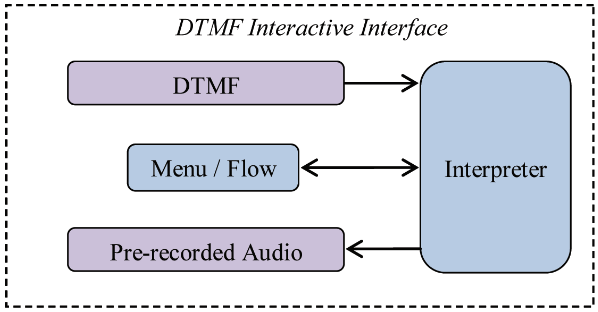
INTERACTING VIA SPOKEN DIALOGUE

The term usability has been used for many years to denote that an application or interface is *user friendly*, *easy-to-use*. It applies to most interfaces, including web interfaces and more importantly speech-based web interfaces, and it can be assessed on both full system level and individual modules and processes level. Therefore, in order to evaluate usability it is important to first understand the design requirements and the architecture of such interfaces. In the following sections we describe the main interaction frameworks that the architecture of most speech enabled applications falls into.

Multimodal Interaction Framework

A general framework (Larson et al., 2003) for the description and discussion of multimodal interaction on the web is developed by the World Wide Web Consortium (W3C). It describes the input and output modes that can be used in a relational abstractive architecture that includes all component types required for the interaction.

Figure 1. Non-speech based interface



In such framework, an application may handle several requests through one or more input modes and respond accordingly. The user may use their input options to make a request for an archive retrieval, the system may respond by either requesting an explicit verification or present all options from the retrieval function, the user may specify or select their preference, allowing the application to present the information. More information on multimodal dialogue can be found in the latest literature (Kuppevelt et al., 2005; Wahlster, 2006).

Non-Speech Based Interaction: DTMF Interactive Systems

DTMF-based interactive systems are widely used for many applications, either web-based or telephone based. The main modalities used on such systems are audio, typing, and point-and-click. Speech is usually pre-recorded since all states of the dialogue are predetermined. The flow is state-based, usually a tree structure flow with options presented to the user at each step. Figure 1 shows the typical design of a non-speech interface where a menu-driven interactive system uses DTMF as input from a series of available choices and responds using predetermined recorded audio.

Such systems may alternatively present the information visually on a screen, thus using the visual modality as well, if the application and design permits.

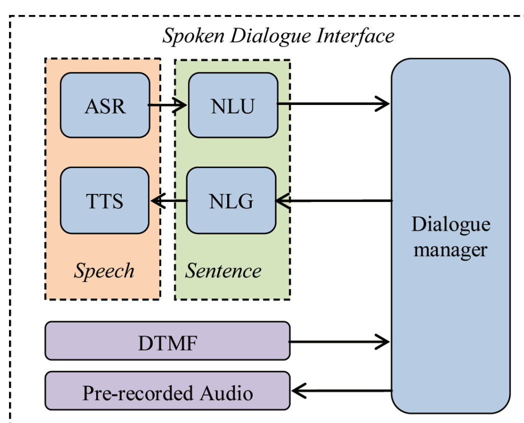
Speech-Based Interaction Framework

The use of speech as input/output for interaction requires a spoken language oriented framework that adequately describes the system processes. W3C has defined the Speech Interface Framework to represent the typical components of a speech-enabled web application (Larson, 2000). A general depiction of a Spoken Dialogue Interface is shown in Figure 2.

A generic dialogue system comprises of the following basic modules:

- The Automatic Speech Recognition (ASR) module that converts user's spoken input into a text string. In addition, a DTMF rec-

Figure 2. Spoken dialog interface



ognizer may be used to allow for DTMF input as well.

- The Natural Language Understanding (NLU) module that interprets the text string passed by the speech recognizer, assigning it an appropriate semantic value.
- The Dialogue Manager (DM), the core of the dialogue system and probably the most complicated component. It handles the conversation flow, evaluating the input and creating the output. In order to do that, it first evaluates and if necessary disambiguates the NLU input based on knowledge about general conversation principles as well as specific conversation context (dialog state and history, task, domain and user information), and then proceeds to create a specific dialog strategy in order to respond. Accordingly, it updates the state of the dialog (or belief state), formulates a dialog plan and employs the necessary dialog actions in order to fulfil the plan. The DM further accesses and utilizes all external knowledge resources, such as back-end databases and world knowledge.
- The Natural Language Generator (NLG) that formulates the actual system prompts, converting the DM output from abstract communicative acts into a well formed written utterance.
- The Text to Speech Synthesizer (TTS) that converts the text passed by the NLG to speech and/or audio. Natural Language Generators are typically coupled with TTS synthesizers but they can also be coupled with Concept-to-Speech (Pan & McKeown, 1997) or Document-to-Audio (Xydas et al., 2004) synthesizers that make use of appropriate linguistic markup and document meta-information in order to manipulate the utterance prosody and achieve increased naturalness and legibility of synthetic speech. Alternatively, in simpler systems, pre-recorded prompts may be used instead. In that case the DM substitutes for the NLG and the TtS forming the output by registering all text prompts and correlating them with prerecorded audio files.

The technologies used for each component may differ depending on the type of the spoken dialogue interface at hand. Following McTear (2004) there are three basic types of spoken dialogue interfaces based on their design and dialogue management techniques involved:

i. State based directed dialogue systems, DTMF replacements
ii. Frame based directed dialogue systems
iii. Agent based, natural language mixed-initiative conversational systems

State-based directed dialogue systems are the very basic menu-driven interfaces where a static tree-based layout is presented to the user. The user may respond with yes/no answers or a limited set of in domain phrases/commands and navigate through the menu options. Such systems are not very efficient nor user-friendly, as the user has to spend precious time going through various levels of menus and listening to every option, in order to complete an often simple task. The main advantage is that they are very robust, posing low recognition and understanding challenges.

Frame-based systems use more advanced techniques in order to accommodate a more natural interaction with the user. The menus may be dynamic, have confirmation and disambiguation prompts as well as more elaborate vocabulary. Furthermore they can support limited mixed initiative dialogue strategies, as they can handle overspecification; that is the user can provide more items of information than those requested by the system at each dialogue turn. Still, user input needs to be properly restricted so that it can be handled by the grammar. On the upside, the small grammars keep the system relatively robust. Such systems are the most commonly used on the market today, providing an industry feasible trade-off between efficiency and robustness.

Agent-based systems are used for large scale applications. These systems are targeted for user satisfaction and naturalness. The users may respond to natural open ended "how may I help you" system prompts with equally natural replies. The utterances may be long, complex and exhibit great variety. The dialogue is dynamic and the demand for successful ASR and NLU is high. Statistical or machine learning methods may be used for interpretation or/and dialogue management as well. The dialogue management is primarily plan-based, the system creating tasks and plans of actions to fulfil. The users expect high-level natural interaction, a very important element to factorise in usability parameterisation.

More on speech-based systems and speech-based interaction enabling technologies can be found in respective textbooks (Dybkjær et al., 2007; Dybkjær & Minker, 2008; Tatham & Morton, 2005, Bernsen et al., 1998; Jurafsky & Martin, 2000; Huang et al., 2001; McTear, 2004).

Voice Browsers

In Robin & Larson (1999) a voice browser is broadly defined as "a device which interprets a voice markup language and generates a dialog with voice output and possibly other output modalities and/or voice input and possibly other modalities". Voice browsers are, by design, single-initiative (system or even user-directed) dialogue applications with a very limited domain and limited dialogue strategy, where dialogue management complexity is not a demand. They are meant to provide the means to browse information and navigate web documents. In our analysis, voice browsers can be considered as a subset of the spoken dialogue web interface description. In this respect, the usability requirements and evaluation methods for spoken dialog web interfaces discussed later in this chapter also apply to voice browsers.

SPOKEN DIALOGUE CHARACTERISTICS

Before entering the usability realm, it is important to first take a look at the inherent characteristics of speech and spoken dialogue in particular as the main interaction modality, since they affect principal design and usability aspects of speech based interfaces compared to non-speech web interfaces. As the underlying philosophy of the web interface designer is directly dependent on the mode of communication, the same service would be designed and implemented in much different way if the hosting platform was a traditional point-and-click web interface than a speech-based one.

First of all, the transient, ephemeral nature of speech along with human cognitive limitations place constraints on the speech output, the amount of information that may be presented to the user and the application structure in general. Non-speech web interfaces, on the other hand, utilize vision and space, and can present a large amount of information that can be easily and quickly processed by the user. Visual menus, which exploit recognition rather than recall, may include up to ten choices per level (Galitz, 2007), while in the case of spoken menus a breadth of three or four information items is recommended (Cohen et al, 2004). As a result, more steps and longer time are often required for speech-based interface users to complete their task. Nor navigating these menus is such a simple task when vision, space and feedback on navigation history are not available. Thus, it comes as no surprise that it is traditional web interfaces – as opposed to speech based ones – that are so strongly menu oriented and information rich. While menus and lists may be a very efficient and useful tool for the former, menu navigation in the case of the latter may result in a significant decrease in efficiency and user satisfaction.

In addition, complex prototypically visual structures such as data tables or nested bulletin are particularly difficult to effectively communicate through speech. Prosody manipulation and use of earcons can improve the "intelligent" acoustic rendition of such structures, as it has been shown by Spiliotopoulos et al (2009a).

Prosody in general is a significant dimension of spoken language that can be utilized to convey discourse and information structure, turn taking protocol, speech act type or even emotions. Final lowering of the fundamental frequency (f0) of the utterance may denote declaration instead of question, high f0 boundaries have been characterized as turn yielding, increase in duration often correlates with emphasis, while rapid changes in intensity may suggest anger. As a result, when prosody is properly manipulated, processing speed increases, comprehension is facilitated and the interaction as a whole becomes more engaging.

Another aspect particularly important to spoken dialogue compared to non speech modalities is grounding. Grounding is the establishment of common ground among the interlocutors. The term refers to the goal and process of achieving mutual understanding within the dialogue and acknowledging this understanding, thus making the other participant confident of the progress made to fulfil the dialogue's goal. In traditional web interfaces the stable visual representation provides the user with the interaction context aiding memory and providing instant feedback on system's actions. Speech based interfaces, in contrast, lack the underlying visual cues and so the system needs to both acknowledge its understanding and remind the users of the interaction point they are at. Yankelovich et al (1995) note that users are often confused when the system does not explicitly acknowledge shared understanding.

Furthermore, several considerations should be addressed with regards to the technology behind speech based interfaces. When building a speech-based human-computer interaction system, certain basic modules must be present. The Dialogue Manager is responsible for the system behavior, control and strategy. The ASR and NLU recognize the spoken input and identify semantic values. The language generator and TtS or the prerecorded audio generator provides the system response. The performance of the particular modules is an indication of usability issues. The ASR accuracy and the lack of language understanding due to dysfluencies, out-of-grammar utterances or ambiguity hinder the spoken dialogue. Moreover, the lack of pragmatic competence of the dialogue manager (compared to the human brain) and the response generation modules sometimes overcomplicate the dialogue and frustrate the user. In the case of graphical interfaces, on the contrary, mouse or keypad input interpretation is more or less error free, and the rest of the components are missing.

Still there are significant advantages to the speech modality. Firstly, speech is an indispensable part of a design for all approach to user interfaces, providing an accessible alternative medium to a number of users such as people with print disability (people with partial or total vision loss, cognitive limitations or limited dexterity) or the elderly. Speech based interfaces can overcome the physical barriers that apply to graphical interfaces making it possible to browse the web, access information and use any application from any telephone anytime and anywhere. Speech is ideal for hands/eyes busy situations such as equipment repairing or driving, or as a browsing alternative on small mobile phone screens. As part of a multimodal interface, it provides users with additional control over the way they interact with the system and express their intent.

Moreover, for several tasks speech is the most efficient medium. People speak at least three times faster than they type (Karat et al., 1999), and multiple actions on a visual display can be performed with a single spoken command. Finally, speech is a more natural means of communication and the one that people are more experienced with. First we learn to talk, and then we learn to read, write or type. Speech based interfaces accommodate conversational behaviour learnt implicitly at a very young age. Graphical interfaces, on the other hand, often resort to use of arbitrary symbols that users are not familiarized with.

All the above considerations affect key usability parameters, some of which are particular to speech based interfaces. For example, low speech recognition and interpretation success rate considerably undermine the user experience (Kamm & Walker, 1997). Voice output quality, feedback adequacy and quick recovery from misrecognitions are only some of the usability aspects that are especially important for speech-based interfaces. Similarly, testing methodology issues may vary.

SPEECH-BASED INTERFACES AND USABILITY

As already mentioned, usability is a broad term that refers to various types of interfaces including graphical web interfaces and speech based web interfaces, denoting that the interface is u*ser-friendly*, *easy-to-use*. Usability is measured according to the attributes that describe it, as explained below (Rubin & Chisnell, 2008):

- Usefulness – measures the level of *task enablement* of the application. As a side result it determines the *will* of the user to actually use it for the purpose it was designed for.
- Efficiency – assesses the *speed*, *accuracy* and *completeness* of the tasks or a user's goal. This is particularly useful for evaluating an interface sub-system since the tasks may be broken down in order to evaluate each module separately.
- Effectiveness – quantifies the system *behaviour*. It is a user-centric measure that calculates whether the system behaves the way the users expect it to. It also rates the system according to the level of *effort* required by the user to achieve certain goals and respective *difficulty*.
- Learnability – it extends the effectiveness of the system or application by evaluating the user's effort required to do specific tasks over several repetitions or time for training and expertise. It is a key measure of user experience since most users expect to be able to use an interface effortlessly after a period of use.
- Satisfaction – it is a subjective set of parameters that the users are asked to estimate and rank. It encompasses the user overall *opinion* about an application based on whether the product meets their *needs* and performs *adequately*.
- Accessibility – in the strict sense, it is not part of the usability description. As a starting point, it is a totally different approach on system design. Accessibility is about access to content, information, and products by everyone, including people with disability. *Design-for-all* is a term that denotes that an application is designed in such way so that everyone can use it to full extent (Stephanidis, 2001). An accessible web site should be implemented according to specification in order to enable voice browsers to navigate through all available information. An accessible web interface should allow for everyone to use. A blind user, for instance, could use certain modalities for input but the system should never respond by non-accessibly visual content (Freitas & Kouroupetroglou, 2008). Accessibility is a very important and broad discipline with many design and implementation parameters. It can be thought as an extension of the aforementioned usability attributes to the universal user. Universal Accessibility (Stephanidis, 2009) strives to use most modalities in order to make the web content available to everyone. Speech and audio interfaces are used for improved accessibility (Fellbaum & Kouroupetroglou, 2008; Duarte & Carriço, 2008). For example,

spoken dialogue systems are considered as key technological factors for the universal accessibility strategies of public terminals, information kiosks and Automated Teller Machines (ATMs) (Kouroupetroglou, 2009). It is mentioned here for completeness; however, it is out of the scope of this chapter.

Interaction Design Lifecycle (Interfaces) and Usability

The basic interaction design process is epitomized by the main activities that are followed for almost every product. There are five activities in the lifecycle of a speech interface (Sharp et al., 2007):

- Requirements specification and initial planning
- Design
- Implementation and testing
- Deployment
- Evaluation

In terms of usability there are three key characteristics pertaining to user involvement in the interaction design process (Sharp et al., 2007):

- User involvement should take place throughout all five stages.
- The usability requirements, goals and evaluation parameters should be set at the start of the development
- Iteration through the four stages is inevitable and, therefore, should be included in the initial planning.

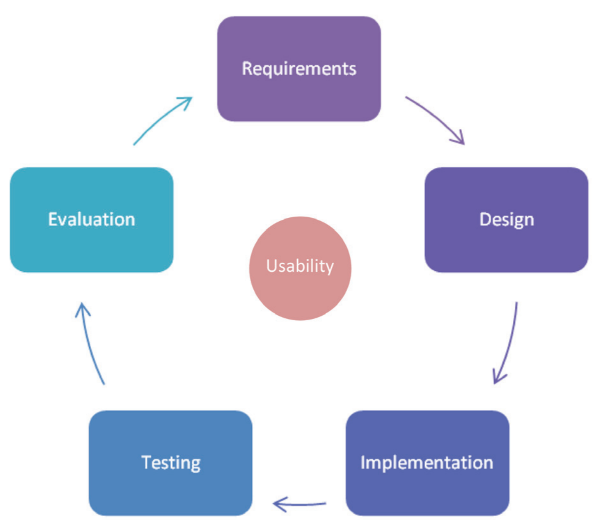
Figure 3 shows how usability generally integrates with the development of a speech-based dialogue interface.

As mentioned in a previous section, spoken dialogue interfaces may be of three basic types (repeated here for ease of presentation):

- State based directed dialog systems, DTMF replacements
- Frame based directed dialog systems
- Agent based, natural language mixed-initiative conversational systems

As each type differs with regards to the level of complexity and sophistication, it becomes clear that each type entails particular usability expectations, and is expected to excel in certain aspects. Table 1 illustrates how usability is taken

Figure 3. Typical interface lifecycle of a speech-based dialogue system



into account in each stage of the product lifecycle depending on the type of speech-based application (Spiliotopoulos & Kouroupetroglou, 2010).

The development of such applications is an iterative process, as mentioned before. Based on our-own experience from our involvement through the development and testing of a number of nationwide-size spoken dialogue business applications, we can declare that practitioners in industrial settings agree that usability parameters, as well as testing, are also part of the iterative process (Kouroupetroglou & Spiliotopoulos, 2009). Agent-based systems possess the highest potential for usability integration. In that respect, the remainder of this chapter refers mostly to agent-based systems and less to the other two. These days, such systems are the centre of the attention by researchers, developers and customers alike, focusing on advanced voice interaction and high user satisfaction. The use of natural voice response (regarding both wording and prosody) and the natural dialogue flow constitute the state-of-the-art in spoken dialogue interfaces. The web provides the means for the application deployment and the system-world communication, aiming to provide stability and vast amount of available information.

Typical Requirements for Real-Life Spoken Dialogue Interfaces

In systems engineering the term *non-functional requirements* is used to denote the requirements that specify the criteria for assessing the operation of a system, while *functional requirements* define the behaviour. In this context, usability is one of the major functional requirements. Non-functional requirements do not encompass usability per se, however they are effective constraints on the design of the system and may indirectly affect the user experience.

Before the start of the design phase, there are certain accustomed typical requirements pertaining to the areas that the design should focus, i.e. the actual issues that the spoken dialogue system is asked to realize or abide with. Some of them are specifically usability-oriented while others are domain-dependent or generic system-oriented. These typical requirements for voiceweb interfaces are:

- User satisfaction – Users that should be satisfied or very satisfied either as stand-alone users or comparing their input from using an earlier interface.
- Quality of service offering – improvement on the quality of the way the requested services/tasks are presented. For example, a large DTMF tree-based dialogue may require the user to navigate through several menu layers to achieve their goal, while a natural language dialogue may identify the initial user request and retrieve the requested service right at the start.
- State-of-the art solution – The system should deploy cutting edge technology.
- Ability to provide customised behavioural or personalised interaction for specific user groups. A common example is the use of a preferred type of interaction (formal, casual, friendly, entertaining, etc.) set specifically for the domain.

Table 1. Usability impact on spoken dialogue interface development lifecycle

Type	Requirements	Design	Implementation	Testing	Evaluation
a	low	medium	low	low	low
b	medium	medium	low	low	medium
c	high	high	medium	high	high

- Complete access to all services or business units that are supported. By design, the system should be able to provide the users the same high quality interaction for all services that the interface is used for.
- Reliability – extends to the system providing the intended functions continuously without failing.
- Continuity of processing – including problem recovery. In this case a natural language interface should cater for the interaction when a system problem occurs.
- Auditability – ensuring the transparency of the system providing supporting evidence to trace processing of data.
- Performance requirements describe the capacity of the system to perform certain functions or process certain volume of transactions within a prescribed time.
- Usability-related factors that the operator of a spoken dialogue interface may find prudent to stress upon to the designer.

These requirements are usually followed by a list of mandatory *acceptance tests* that the final system should pass before it is deployed to the web. The format imposed for the acceptance tests is generally comprised of *key performance indicators* (KPIs) for ASR and TtS success. These should be developed by the designer and be available on production to use also for tuning purposes. Furthermore, acceptance tests include task completion evaluation for all requested tasks that are to be tested.

For average size/complexity spoken dialogue interfaces, a magnitude of 10-15 trialists should be sufficient for the acceptance tests. There are two main areas that the tests are carried out in:

A. *Functional assessment* of the system respective modules and functions such as accuracy of information relayed to the user, start/end of dialogue or sub-dialogue flow, service/information provision accuracy, and so on.

B. *User Experience assessment* in terms of
   - B.1 quality issues
     - B.1.1 speech or dialogue pause length between activities such as voice request, system search, information retrieval, information relay/output, and prompt delays between responses
     - B.1.2 output voice (natural or synthetic) consistency and naturalness for all stages of dialogue as well as in special cases where critical information or explicit help is required
     - B.1.3 choice of presenting output voice, clear and non-breaking, during loudspeaker mode or in noisy environments
     - B.1.4 correct pronunciation and focus placement in sentences
   - B.2 user interaction
     - B.2.1 ease of use, navigation through the interface
     - B.2.2 instructions and help prompt quality
     - B.2.3 smart recovery from misinterpretations or misrecognitions
     - B.2.4 disambiguation and confirmation function performance
     - B.2.5 dialogue flow cohesion
     - B.2.6 overall satisfaction.

Since all this information is available to the designer beforehand, it can be put to good use especially during the design. Most of these requirements are the constraints set by the operator so that the design should be built around them. A good design should take those into account in order for the final system to pass the acceptance test assessments.

USABILITY EVALUATION FOR SPEECH-BASED SYSTEMS

Usability evaluation can be formative or summative and thus it can be performed either during or at the end (or near the end) of the development cycle. The methodologies that can be used for that differ in their scope, their main difference being that, when a product is finished (or nearly finished), *usability testing* serves for fine-tuning certain parameters and adjusting others to fit the target user better. During the design phase, usability evaluation methods can be used to probe the basic design choices, the general scope and respective task analysis of a web interface. Some of the most common factors to think about when designing a usability study are:

- Simulate environment conditions closely similar to the real world application use.
- Make sure the usability evaluation participants belong to the target user group
- Make sure the user testers test all parameters you want to measure
- Consider onsite or remote evaluation.

These factors are referenced later in this section.

Methodologies

Usability evaluation for speech-based web interfaces is carried upon certain evaluation methods and approaches on the specific modules and processes that comprise each application. Each approach measures different parameters and goals and can be applied at different stages in the product lifecycle. They all have the same main objective, to evaluate usability for a system, sub-system or module. However, each approach targets specific parameters for evaluation. The main two usability evaluation classes for spoken dialogue systems include the Wizard-of-Oz (WOZ) formative testing (Harris, 2005) and the summative usability testing.

The Wizard-of-Oz Formative Evaluation

It is a common formative approach that can be used not only for speech-based dialogue systems but for most web applications. It allows for usability testing during the early stages by using a human to simulate a fully working system. In the case of speech-based dialog systems, the human "wizard" performs the speech recognition, natural language understanding, dialog management and output generation. The WOZ approach is primarily used during the initial design phase to test the proposed dialogue flow design and the user response to information presentation parameterisation. Main advantages of the approach are listed below (Cohen et al., 2004; Harris, 2005):

- Early testing – it can be performed in the early stages in order to test and formulate the design parameters as early in the product lifecycle as possible. That way design shortcomings that would be costly to fix later in the product's lifecycle are avoided.
- System updates – the system, being a mock-up, can be updated effortlessly to accommodate for changes imposed from the input from the test subjects, making it easier to re-test the updated system in the next usability evaluation session.
- Language resources - Grammar coverage for the speech recognition (ASR) and respective machine learning approaches for interpretation (NLU) are always low when testing a non-finalised product, and as such they may hinder the usability evaluation. However, the use of the human usability expert eliminates such handicap. Moreover, problems such as integration bugs that may arise later in the development are eliminated in the case of the mock-up.
- Significant information can be gathered about the vocabulary and the syntax used, the users' attitudes and mental model of the task. The dialogues collected can also

be utilized as initial training corpus for the implementation of the ASR and NLU components.

On the other hand, the WOZ approach faces certain drawbacks that are inherent to user-testing in general:

- Unrealistic system use conditions – as test participants are not motivated in the same way as real users are, and are often not representative of the end user population. Earlier studies (Turunen et al., 2006; Ai et al., 2007) have shown that there are differences between usability testing and actual use conditions; differences refer to the use of barge-in, explicit help requests, significant silence timeouts, speech recognizer rejection rates and dialog duration among others. For WOZ testing in particular, the realistic aspect is further compromised, as speech recognition and language interpretation errors are difficult to simulate and not taken into account.
- User bias – the language used to describe the tasks to be performed inevitably influences the participants' choice of vocabulary and utterance structure, and so the utility and reliability of elicited language patterns and behaviour are undermined.
- Resources consumed – setting up a WOZ experiment requires people and time, as well as tools that can be costly to develop (Jankelovich, interview in Weinschenk, 2000)

There are two requirements for successful usability testing, the design of the tasks and the selection and training of participants. The participants must be representative of the end-user population, taking into account age, demographics, education. Other criteria may be set depending on the actual application domain, for example users of a specific web site. Moreover, novice and expert users can be recruited in order to provide the means of applying the system design to the worst-case (low experience level) and best-case (high experience level) population.

The participants are required to complete a number of tasks that are carefully selected to test the system. In a dialogue system the primary concern to evaluate is the dialogue flow. Two sets of scenarios should be designed, one asking the participants to perform specific actions or pursue predetermined goals using scenarios and another asking for uncontrolled access of the system pursuing goals of their own choice. The controlled predetermined scenarios are used to evaluate the behaviour of the participants against the expected behaviour of the designer, exposing possible flaws of the design. The uncontrolled interaction is used to evaluate the generic performance of the participants revealing the basic faults of the design, such as non-obvious availability of *help* function or *ambiguous* interaction responses from the system.

The results of the WOZ tests are both from the user subjective feedback and the examination of the objective performance measures. The performance measures include:

- task completion – whether the participants completed the specified tasks that were set within the scenarios successfully,
- efficiency – whether the participants chose the most direct route to the goal, using the predetermined scenario feedback to compare against the optimal path for the same scenario that was expected by the designer,
- dialogue flow – how the participants chose to interact with the system,, the number of times the help was requested and how informative it was, as well as the number of times disambiguation, confirmation and error recovery sub-dialogs were enabled.

The subjective input of the participants is recorded through questionnaires that the participants fill in after each task completion as well as at the end of the evaluation. The questions are used to assess the user experience asking about complexity, effort required, efficiency, linguistic clarity, simplicity, predictability, accuracy, suitable tempo, consistency, precision, forgiveness, responsiveness (see Ward & Tsukahara, 2003), appropriateness, overall impression and acceptance of the system either regarding particular tasks or the full system (Weinschenk & Barker, 2000). The participants are usually asked to mark their level of their agreement to the questions through a 1 to 5 or 1 to 7 scales (for example, 1 being "totally disagree" and 7 "totally agree" and the rest in between), commonly known as Likert scales.

The data are analysed and problems are prioritized in terms of type, severity, and frequency. The subjective feedback also indicates behavioural flaws in the design. Both results enable the designer to take certain action to fix or eliminate those flaws from the design and proceed to implementation.

The Summative Usability Testing of Voice Web Systems

At the end of the implementation, pre-final versions of the system should be tested by potential users in order to evaluate the usability. Usability testing at this stage is not much different to WOZ in terms of planning. But now there is no human actor (wizard) but the full system interaction with the user. This means that the ASR, context interpretation and generation, and TtS are now part of the usability metrics.

At this stage, the usability tests play a much more pivotal role since the development of the system is near completion. There are three distinct purposes for usability testing of a working system: the *development*, *testing* and *tuning*. During the development the users test a nearly finished product, during testing a finished product, and during tuning a finished and already deployed product. Regardless of purpose, the tests focus on all aspects that the WOZ handled as well as several aspects that the WOZ ignored:

- Grammar testing
- Interpretation testing
- Dialogue management/flow
- System response adequacy
- Output speech quality.

For spoken dialogue interfaces, the following 15 objective (both quantitative and qualitative) and subjective usability evaluation criteria have been proposed (Dybkjær & Bernsen 2000):

1. Modality appropriateness.
2. Input recognition adequacy.
3. Naturalness of user speech relative to the task(s) including coverage of user vocabulary and grammar.
4. Output voice quality.
5. Output phrasing adequacy.
6. Feedback adequacy.
7. Adequacy of dialogue initiative relative to the task(s).
8. Naturalness of the dialogue structure relative to the task(s).
9. Sufficiency of task and domain coverage.
10. Sufficiency of the system's reasoning capabilities.
11. Sufficiency of interaction guidance (information about system capabilities, limitations and operations).
12. Error handling adequacy.
13. Sufficiency of adaptation to user differences.
14. Number of interaction problems (Bernsen et al. 1998).
15. User satisfaction.

Bernsen & Dybkjær (2000) have have proposed the use of the *evaluation templates*, i.e. "models of what the developer needs to know in order to apply an evaluation criterion to a particular property of a Spoken Language Dialogue System or component", in their methodology as best practice guides. Later, they formed a set of guidelines for up-to-date spoken dialogue design, implementation and testing, covering seven major aspects: informativeness, truth and evidence, relevance, manner, partner asymmetry, background knowledge, repair and clarification (Bernsen & Dybkjær, 2004). These aspects can be used as the basis for usability testing strategies and for evaluation frameworks (Dybkjær & Bernsen, 2001; Dybkjær et al., 2004). One of them is the PARADISE evaluation framework (Walker et al., 1998; Hajdinjak & Mihelic, 2006) with general models developed for it (Walker et al., 2000).

As with WOZ, usability testing needs participants. The recruitment procedure is pretty much the same as described earlier in WOZ, with a few additional parameters. The participants use the real system, which means that, at this stage, functional parameters in speech recognition and speech synthesis should be tested, measured and decided upon. As mentioned above there is extensive work on the comparison of usability evaluation feedback between in-house recruited participants versus real users indicating substantial differences. Moreover, parameter measurements in speech recognition rejection, choice of interaction ending, help and repeat requests, user interruptions and silence timeouts, show that there users behave differently in the first month of their interaction. After that, the users become accustomed to the system, experienced and their behaviour becomes more or less stabilised (Turunen et al., 2006).

Kamm et al. (1998) stress the importance of a successful quick tutorial on the users before using a speech-based application. They show that the initial user experience can be ensured when the first-time users are trained on the use of the system. The user satisfaction and the system performance were significantly improved in this case. Also, there is significant differentiation between onsite and remote evaluation. Participants recruited for onsite evaluation know that they are required to evaluate the system and may behave unexpectedly or even use extreme caution when using the system, a behavior much dissimilar to that of real users.

Apart from task completion and dialogue flow, depending on the domain, as a general rule, functional measurements should be recorded for at least the following indicative parameters:

- Average call duration
- Peaks and valleys of usage per hour per day
- Successful speech recognitions
- Misrecognitions
- No-inputs
- Timeouts
- Rejections
- Early hang-ups
- Successful interpretations
- Failed interpretations (no-matches)
- Successful repairs
- Failed repairs.

Performance metrics may be derived by calculating parameters such as the number of user and system turns, elapsed time, number of help requests, number of timeout prompts, mean ASR accuracy and number of speech recognition rejections (Kamm & Walker, 1997; Kamm et al., 1999). Generally, the above parameters can indicate functional problems with the application and the degree that each of those affects the user experience (Walker et al., 1999). Furthermore, the data can be automatically processed using appropriate methods (Hartikainen et al., 2004), used to train models for evaluation-based problem prediction that leads to an adaptive spoken dialogue interface (Litman et al., 1998; Litman & Pan, 2002).

The user subjective feedback is also very important at this stage. It illustrates the user experience as perceived by the user (to be compared with automatically-derived user experience level form the performance metrics analysis) and stresses the points where the users were not satisfied. By analysing the questionnaires down to the usability factors (Larsen, 2003) the designer can even, to an extent, predict the quality and usability of spoken dialogue services (Moller et al. 2006; 2008).

Usability Challenges for Advanced Speech-Enabled Web Interfaces

VoiceWeb may include non-speech enabled interfaces for certain tasks that are simple and linear. The aim for choosing such approach is robustness. Lee & Lai (2005) have made a comparative study on the use of speech-enabled versus DTMF approaches and reported on usability issues. The scope was limiting enough to implement via a DTMF system. The results showed that for the specific tasks the effectiveness and efficiency (summative evaluation) of the DTMF system were clearly much higher than the speech-enabled one for all but the sole most complex task. That task required longer communication between the user and the system, which in effect favoured the spoken dialogue. The users, however, in their formative evaluation clearly preferred the speech modality and the respective system, in terms of naturalness, ease-of-use, and overall satisfaction. That was also the case for the usability evaluation comparison between a DTMF and a speech-enabled interface for a large customer care domain. The complexity of the domain showed the clear advantage of the spoken dialogue system in terms of system and dialogue/task level criteria such as usefulness, efficiency, naturalness, user satisfaction, system behaviour, interaction flow and initiative, etc. (Spiliotopoulos et al., 2009b)

The above study illustrates and proves the need for speech-enabled communication for all but the very simple tasks. Earlier studies (Delogu et al., 1998) have indicated that the usability evaluation methodologies should be adapted to the specific complexity of each approach or system that is evaluated, while others provided extensive sets of objective and subjective measures that can be applied for specific measurements for usability evaluation of spoken dialogue systems (Larsen, 2003).

This brings the topic to a new level, the determination of how a web interface can best cope with the speech modality depending on the complexity of the tasks and the type of information that must be communicated. In terms of application domain and requirements, there are two major types of applications:

i. *The standard, widely-used speech based applications,* such as call centres, customer care automated spoken dialogue systems.
ii. *The complex, speech-enabled approaches for interfacing composite information that requires intense navigation and/or delicate communication sub-tasks.* Such approaches can be used when a system must present many pieces of compound information to the user according to the user feedback, and re-adjust, interrupt and re-focus the search or delivery of information as needed.

The latter presents a challenge when trying to render complex information to the speech modality. For example, rendering the data and their respective relations from a normal table structure to speech is a very hard task that requires extensive manipulation of factors such as prosody or auditory markers (Spiliotopoulos et al. 2010). Moreover, the requirements of the spoken output and the ability of the system to confirm that the user understood the spoken output (confirmation, disambiguation) are quite different and much more demanding. The NLG and TTS components required for such tasks are more refined and include both grammars and adaptive statistical approaches, as do the NLU and Dialogue Management. There are certain semantic relations, sometimes in the form of meta-information that must be relayed to the user in order to make the output understandable.

All these considerations must unavoidably be integrated to the usability evaluation for such systems. The suitable usability evaluation criteria based on their significance throughout the development lifecycle process are denoted below:

A. Output quality
- A.1 Dialogue-related TTS utterances – statement, confirmation, disambiguation
- A.2 Content sentence generation – NLG
- A.3 Quality of TTS – prosody, clarity of information provision

B. Spoken Input
- B.1 User input handling – NLU
- B.2 Dialogue management decisions based on successful NLU

C. System parameters
- C.1 Response time
- C.2 Coherence and continuation of information
- C.3 Low deviation of the mean number of turns needed for task completion
- C.4 Number and type of misrecognition or rejections of user input

D. Task completion
- D.1 Percentage of task completion
- D.2 Analysis of incomplete tasks by examining the process and identifying the parameters from groups 1, 2 and 3.

The key to evaluating such systems is the determination of how each criteria and respective parameters affect the system efficiency and user acceptance. In order to do that a combined method of WOZ testing and early system parameterization can successfully formulate particular performance metrics bases for the later testing and evaluation.

CONCLUSION

Advances in speech processing, natural language understanding and web technologies have allowed the development of useful, universally usable and user friendly speech enabled web interfaces. Utilizing the speech modality for web applications can increase naturalness, efficiency and user satisfaction, provided that usability issues and parameters are taken into consideration and usability evaluation is integrated throughout the interface's lifecycle. In this regard, this chapter outlined the different types of speech based interaction and the ways in which speech based interfaces differ from traditional point-and-click web interfaces posing particular usability challenges. It presented usability evaluation methods and approaches focusing on usability frameworks primarily targeted for speech-based web interfaces. Furthermore, usability challenges presented by complex speech based applications in terms of output – as in the case of data tables rendition – or high NLU demands were discussed. Such issues are important to factorize in usability evaluation to ensure the development of high quality applications.

In general, it is important that usability testing is employed in all steps of the process, from design to deployment and maintenance, and users' input is accordingly taken into account. First, the designer utilizes user feedback to formulate an interface that provides all the requested functionality in a user-friendly, user-tested and approved manner. Throughout the implementation phase the system is put to test and design choices are further validated. Once the system is finished or nearly finished, specified functional tests are performed and usability evaluation assures that requirements are met. Finally, the already deployed system is re-evaluated for quality assuring or updating purposes. For the latter a WOZ-like evaluation can be used that enables the developers to intervene during the interaction and try out alternative flows and presentation modes. In short, building a speech-based interface is an ongoing, iterative test and development process with more general as well as particular to the speech modality usability objectives, methods and approaches. The development of and abiding by advanced usability standards becomes even more imperative, as web interfaces become more sophisticated allowing for more flexible and natural spoken language input and output.

ACKNOWLEDGMENT

The work described in this chapter has been funded by the Special Account for Research Grants of the National and Kapodistrian University of Athens under the KAPODISTRIAS program.

REFERENCES

Ai, H., Raux, A., Bohus, D., Eskenazi, M., & Litman, D. (2007). Comparing spoken dialog corpora collected with recruited subjects versus real users. In Proc. of the 8th SIGdial workshop on Discourse and Dialogue, pp. 124–131.

Bernsen, N. O., Dybkjaer, H., & Dybkjaer, L. (1998). *Designing Interactive Speech Systems: From First Ideas to User Testing*. New York: Springer-Verlag.

Bernsen, N. O., & Dybkjær, L. (2000). A Methodology for Evaluating Spoken Language Dialogue Systems and Their Components. In Proc. 2nd International Conference on Language Resources & Evaluation - LREC 2000, pp.183-188.

Bernsen, N. O., & Dybkjær, L. (2004). Building Usable Spoken Dialogue Systems: Some Approaches. *Sprache und Datenverarbeitung*, *28*(2), 111–131.

Cohen, M., Giancola, J. P., & Balogh, J. (2004). *Voice User Interface Design*. Boston, MA: Addison-Wesley Professional.

Delogu, C., Di Carlo, A., Rottundi, P., & Sartori, D. (1998). Usability evaluation of IVR systems with DTMF and ASR, 5th International Conference on Spoken Language Processing, paper 0320, Australia,1998.

Duarte, C., & Carriço, L. (2008). Audio Interfaces for Improved Accessibility. In Pinder, S. (Ed.), *Advances in Human Computer Interaction* (pp. 121–142). Vienna, Austria: I-Tech Education and Publishing.

Dybkjær, L., & Bernsen, N. O. (2000). Usability Issues in Spoken Language Dialogue Systems. *Natural Language Engineering*, *6*(3-4), 243–272. doi:10.1017/S1351324900002461

Dybkjær, L., & Bernsen, N. O. (2001). Usability Evaluation in Spoken Language Dialogue Systems. In. Proc.ACL Workshop on Evaluation Methodologies for Language and Dialogue Systems, pp.9-18.

Dybkjær, L., Bernsen, N. O., & Minker, W. (2004). Evaluation and Usability of Multimodal Spoken Language Dialogue Systems. *Speech Communication*, *43*(1-2), 33–54. doi:10.1016/j.specom.2004.02.001

Dybkjær, L., Hemsen, H., & Minker, W. (Eds.). (2007). *Evaluation of Text and Speech Systems*. Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-1-4020-5817-2

Dybkjær, L., & Minker, W. (Eds.). (2008). *Recent Trends in Discourse and Dialogue*. Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-1-4020-6821-8

Fellbaum, K., & Kouroupetroglou, G. (2008). Principles of Electronic Speech Processing with Applications for People with Disabilities. *Technology and Disability*, *20*(2), 55–85.

Freitas, D., & Kouroupetroglou, G. (2008). Speech Technologies for Blind and Low Vision Persons. *Technology and Disability*, *20*(2), 135–156.

Galitz, W. O. (2007). *The Essential Guide to User Interface Design*. Wiley Publishing, Inc.

Hajdinjak, M., & Mihelic, F. (2006). The PARADISE evaluation framework: Issues and findings. *Computational Linguistics*, *32*(2), 263–272. doi:10.1162/coli.2006.32.2.263

Harris, R. A. (2005). *Voice Interaction Design: Crafting the New Conversational Speech Systems*. San Francisco: Morgan Kaufmann.

Hartikainen, M., Salonen, E.-P., & Turunen, M. (2004). Subjective Evaluation of Spoken Dialogue Systems Using SERVQUAL Method. In Proc. 8th International Conference on Spoken Language Processing - ICSLP, pp. 2273-2276.

Huang, X., Acero, A., & Hon, H.-W. (2001). *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. New Jersey: Prentice Hall PTR.

Jurafsky, D., & Martin, J. H. (2008). *Speech and Language Processing. An Introduction to Natrural Language Processing, Computational Linguistics, and Speech Recognition*. New Jersey: Prentice-Hall.

Kamm, C. A., Litman, D. J., & Walker, M. A. (1998). From novice to expert: The effect of tutorials on user expertise with spoken dialogue systems. In Proc. 5th International Conference on Spoken Language Processing - ICSLP.

Kamm, C. A., & Walker, M. A. (1997). Design and Evaluation of Spoken Dialogue Systems. In Proc. IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 14–17.

Kamm, C. A., Walker, M. A., & Litman, D. J. (1999). Evaluating spoken language systems. In Proc. Applied Voice Input/Output Society Conference - AVIOS, pp. 187–197.

Karat, C. M., Halverson, C., Horn, D., & Karat, J. (1999). Patterns of entry and correction in large vocabulary continuous speech recognition systems. ACM Conference on Human Factors in Computing Systems. Pittsburgh, PA.

Kouroupetroglou, G. (2009). Universal Access in Public Terminals: Information Kiosks and Automated Teller Machines (ATMs). In Stephanidis, C. (Ed.), *The Universal Access Handbook* (pp. 761–780). Boca Raton, Florida: CRC Press. doi:10.1201/9781420064995-c48

Kouroupetroglou, G., & Spiliotopoulos, D. (2009). Usability Methodologies for Real-Life Voice User Interfaces. *International Journal of Information Technology and Web Engineering*, *4*(4), 78–94. doi:10.4018/jitwe.2009100105

Larsen, L. B. (2003). Issues in the Evaluation of Spoken Dialogue Systems using Objective and Subjective Measures. In Proc. 8th IEEE Workshop on Automatic Speech Recognition and Understanding -ASRU, pp. 209-214.

Larsen, L. B. (2003). On the Usability of Spoken Dialogue Systems. PhD Thesis, Aalborg University.

Larson, J. A. (2000). Introduction and Overview of W3C Speech Interface Framework, Retrieved August 2, 2009, from http://www.w3.org/ TR/voice-intro/

Larson, J. A., Raman, T. V., & Raggett, D. (2003). W3C Multimodal Interaction Framework, Retrieved August 2, 2009, from http://www.w3.org/TR/mmi-framework/

Lee, K. M., & Lai, J. (2005). Speech Versus Touch: A Comparative Study of the Use of Speech and DTMF Keypad for Navigation. [Lawrence Erlbaum Associates.]. *International Journal of Human-Computer Interaction*, *19*(3), 343–360. doi:10.1207/s15327590ijhc1903_4

Litman, D. J., & Pan, S. (2002). Designing and evaluating an adaptive spoken dialogue system. *User Modeling and User-Adapted Interaction*, *12*(2-3), 111–137. doi:10.1023/A:1015036910358

Litman, D. J., Pan, S., & Walker, M. A. (1998). Evaluating Response Strategies in a Web-Based Spoken Dialogue Agent. In Proc. 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conf. on Computational Linguistics (ACL/COLING), pp.780–786.

McTear, M. F. (2004). *Spoken Dialogue Technology: Towards the Conversational User Interface*. London: Springer-Verlag.

Moller, S., Engelbrecht, K., & Schleicher, R. (2008). Predicting the quality and usability of spoken dialogue services. *Speech Communication*, *50*(8-9), 730–744. doi:10.1016/j.specom.2008.03.001

Moller, S., Englert, R., Engelbrecht, K., Hafner, V., Jameson, A., Oulasvirta, A., et al. (2006). MeMo: Towards Automatic Usability Evaluation of Spoken Dialogue Services by User Error Simulations. In Proc. 9th International Conference on Spoken Language Processing - ICSLP, pp. 1786-1789.

Pan, S., & McKeown, K. (1997). Integrating language generation with speech synthesis in a concept to speech system. In Proceedings of ACL/EACL'97 Concept to Speech Workshop, Madrid, Spain.

Robin, M., & Larson, J. (1999). Voice Browsers: An introduction and glossary for the requirement drafts. W3C Working Draft, December 1999

Rubin, J., & Chisnell, D. (2008). *Handbook of Usability Testing: Howto Plan, Design, and Conduct Effective Tests*. Indianapolis, Indiana: Wiley Publishing, Inc.

Sharp, H., Rogers, Y., & Preece, J. (2007). *Interaction Design: Beyond Human-Computer Interaction*. West Sussex, England: John Wiley & Sons, Inc.

Spiliotopoulos, D., & Kouroupetroglou, G. (2009). *Usability Methodologies for Spoken Dialogue Web Interfaces. Chapter in the book: Integrating Usability Engineering for Designing the Web Experience: Methodologies and Principles, 2009, Information Science Reference Press*. Pennsylvania, USA: IGI Global.

Spiliotopoulos, D., Stavropoulou, P., & Kouroupetroglou, G. (2009a). *Acoustic Rendering of Data Tables using Earcons and Prosody for Document Accessibility. Lecture Notes on Artificial Intelligence 5616* (pp. 587–596). Universal Access in HCI, Springer Berlin Heidelberg.

Spiliotopoulos, D., Stavropoulou, P., & Kouroupetroglou, G. (2009b). *Spoken Dialogue Interfaces: Integrating Usability. Lecture Notes in Computer Science 5889* (pp. 484–499). HCI and Usability for e-Inclusion, Springer Berlin Heidelberg.

Spiliotopoulos, D., Xydas, G., Kouroupetroglou, G., Argyropoulos, V., & Ikospentaki, K. (2010, June). Auditory Universal Accessibility of Data Tables using Naturally Derived Prosody Specification. [Springer Berlin Heidelberg.]. *Univ. Access Inf. Soc.*, *9*(2), 169–183. doi:10.1007/s10209-009-0165-0

Stephanidis, C. (Ed.). (2001). *User Interfaces for All: Concepts, Methods and Tools*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Stephanidis, C. (Ed.). (2009). *The Universal Access Handbook*. Boca Raton, Florida: CRC Press.

Tatham, M., & Morton, K. (2005). *Developments in Speech Synthesis*. West Sussex, England: John Wiley & Sons, Inc.doi:10.1002/0470012609

Turunen, M., Hakulinen, J., & Kainulainen, A. (2006). Evaluation of a Spoken Dialogue System with Usability Tests and Long-term Pilot Studies: Similarities and Differences. In Proc. 9th International Conference on Spoken Language Processing - INTERSPEECH pp. 1057-1060.

van Kuppevelt, J., Dybkjær, L., & Bernsen, N. O. (Eds.). (2005). *Advances in natural multimodal dialogue*. Dordrecht, The Netherlands: Springer. doi:10.1007/1-4020-3933-6

Wahlster, W. (Ed.). (2006). *SmartKom: Foundations of Multimodal Dialogue Systems*. Berlin, Heidelberg: Springer-Verlag. doi:10.1007/3-540-36678-4

Walker, M. A., Borland, J., & Kamm, C. A. (1999). The utility of elapsed time as a usability metric for spoken dialogue systems. In Proc. IEEE Automatic Speech Recognition and Understanding Workshop - ASRU, pp. 317–320.

Walker, M. A., Kamm, C. A., & Litman, D. J. (2000). Towards developing general models of usability with PARADISE. *Natural Language Engineering*, *6*(3-4), 363–377. doi:10.1017/S1351324900002503

Walker, M. A., Litman, D. J., Kamm, C. A., & Abella, A. (1998). Evaluating spoken dialogue agents with PARADISE: Two case studies. *Computer Speech & Language*, *12*(3), 317–347. doi:10.1006/csla.1998.0110

Ward, N., & Tsukahara, W. (2003). A Study in Responsiveness in Spoken Dialog. *International Journal of Human-Computer Studies*, *59*, 603–630. doi:10.1016/S1071-5819(03)00085-5

Weinschenk, S., & Barker, D. T. (2000). *Designing effective speech interfaces*. New York: John Wiley & Sons, Inc.

Xydas, G., Argyropoulos, V., Karakosta, T., & Kouroupetroglou, G. (2004). An Open Platform for Conducting Psycho-Acoustic Experiments in the Auditory Representation of Web Documents, Proc. of the Conf. ACOUSTICS 2004, 27-28 Sept. 2004, Thessalonica, pp. 157-164

Yankelovich, N., Levow, G. A., & Marx, M. (1995). *Designing Speech Acts: Issues in speech user interfaces. Human Factors in Computing Systems* (pp. 369–376). Denver: Association of Computing Machinery.