Creating Effective, Efficient and Desirable Voice Enabled Web Interfaces

Bryan Duggan

School of Computing, Dublin Institute of Technology,
Kevin St., Dublin 8, Ireland

bryan.duggan@comp.dit.ie
http://www.comp.dit.ie/bduggan

Abstract. The application of principles and guidelines for interaction design are essential in all human-computer interface modalities. This paper outlines a number of important considerations in building systems that use VoiceXML to generate speech interfaces, accessible over the telephone. The considerations outlined in this paper have been categorised based on commonly accepted criteria into effectiveness or usefulness factors, efficiency or usability factors and user satisfaction or desirability factors. The paper also contains a discussion on the importance of branding in voice enabled web systems and presents a taxonomy of factors that influences brand communication. A simple checklist called RAF-HCI is presented based on the authors work developing REVIEW (Roadmap for Enterprise Voice Enabled Web) and RAF (Review Achievement Framework). An industry case study is also presented which analyses a project by the criteria outlined in RAF-HCI.

1. Introduction

The voice enabled web combines XML based mark-up languages (VoiceXML, SRGS, SSML and SALT) speech recognition, text to speech (TTS) and web technologies. It offers the promise of allowing callers to access web based services from any telephone, making it practical to access the web anytime and anywhere, whether at home, on the move, or at work [W3C 2004]. Deployments of voice enabled web technology have proliferated in recent years. Applications such as portals, banking systems, stock-trading systems, insurance and healthcare systems allow callers to interact with web based systems over the telephone by speaking and listening to speech.

Like all human-computer interaction technologies, the success of a voice enabled web system hinges on acceptance by the people who use it. The application of principles and guidelines for interaction design are essential in all human-computer interface (HCI) modalities [Fitzpatrick 98]. A poorly considered user interface for a voice enabled web application will surely lead to the systems failure.

The ISO (International Standards Organisation) in the ISO 9241 - Part 11 standard and the HFES (Human Factors and Ergonomics Society) in the HFES 200 standard have proposed three core criteria, which are critical for the design of any human-computer interface [ISO 98] [HFES 01]:

- Effectiveness or usefulness is the accuracy and completeness which specified users can achieve specified goals in particular environments.
- Efficiency or usability describes the resources expended in relation to the accuracy and completeness of goals achieved.
- User satisfaction or desirability describes the comfort and acceptability of the work system to its users and other people affected by its use.

This paper presents an overview of the key HCI design factors for voice enabled web systems. It categorises these factors according to the criteria outlined by both the ISO and the HFES as outlined above. The research presented in this paper is based on the authors experience developing REVIEW (Roadmap for Enterprise Voice Enabled Web) and RAF (Review Achievement Framework). REVIEW is a seven phase project plan for implementing a voice enabled web system. It covers the entire project lifecycle from initial requirements gathering through to post implementation and maintenance. RAF is an evaluation framework that assesses projects during each phase to measure how closely the roadmap was followed. REVIEW and RAF were developed based on:

- Prototype development and trials.
- Industry consultation.
- Industry case studies.
- Literature review.
- Undergraduate final year project supervision.

The relevant sections from RAF relating to HCI are also presented later in the paper. Further information on REVIEW and RAF can be found at [Duggan 03b].

2. Effectiveness or usefulness factors

These factors address the basic principle that the interface must be functional enough so that a caller can complete the task required. Addressing these issues ensures that the interface is functional for the majority of callers.

Recognition and text to speech engines must be tailored for particular regions to be effective [Markowitz 96]. This is to take account of different accents, phrasing and ways of speaking. Telephone based systems are usually speaker-independent and hence they must recognise large heterogeneous populations of speakers. These models are consequently more complex and difficult to construct than those created for speaker-dependant recognition [Markowitz 96]. Systems will sometimes avoid confusable vocabulary items in order to improve recognition rates [Sharma 02] however this may not be possible with some systems. For example a brokerage system will need to recognise many thousands of company names [Economist 01].

Sampling is often carried out to produce a speech model that is tuned to the target population and the speaking environment of the application [Glass 99]. Often the population will be partitioned into subgroups based on demographic information about age, sex and dialect patterns. Data collection from sub groups reflects the population proportion of that group [Markowitz 96].

Technical solutions such as neural networks can minimise the amount of sampling that needs to be done in order to accurately model the speech patterns of a diverse group of speakers. For more information on the topic of neural networks in speech recognition, readers can refer to [Yuk 99].

If a population contains large numbers of non-native speakers then these must also be included in the recognition model [Van 01]. Consider an Irish person speaking French as opposed to a Parisian speaking French. Speech systems designers often divide users into "lambs" and "goats" [Van 01]. "Lambs" are the majority of users whose speech can be accurately modelled, as their speech patterns are relatively similar. "Goats" are people whose speech patterns are difficult to capture. People can be classified as goats for a number of reasons, but often because they simply have unusual vocal characteristics [Van 01]. For “goats” short-term adaptation can be a possible solution. This involves the system learning from its mistakes with a particular user. This information may then be held in a profile, for when the user next uses the system [Zue 01].

Most commercial platforms ship with recognition engines tuned for a particular population. Nuance Corp for example, ships separate English language recognition engines for the UK, the USA, Australia, South Africa and Singapore [Nuance 04].

Users need feedback from the system so that they understand the state of the interaction. Unlike Graphical User Interfaces (GUI), voice interfaces are invisible – there are no visual cue’s such as graphics, fonts and tables to structure the content. In a GUI environment, fonts, graphics, icons and layout are used to connect users to the content and to help users understand the context of an application [Horton 94]. In a speech environment, speech and non-speech cues can be used to provide structural and context cues. In speech user interface design, these cues are known as *earcons* [Sharma 02]. It is recommended that tones should be short in duration (between 0.5 and 1.0 seconds) so as not become distracting and obtrusive. Earcons can also be used to eliminate dead air by giving users audio feedback that the system is busy. Table 1 lists guidelines for the use of earcons in an application.

Table 1. Guidelines for the use of earcons [Sharma 02]

Earcon	Recommendation
Prompting tone	Prompting tones can be used to indicate to the user that it is their turn to speak or provide input. This can be useful in avoiding caller confusion about when they have to provide input to the service.
Turn-taking tones	Unique tones to indicate that it’s the user’s turn to speak/interact.
Disabled barge-in	During messages such as legal notices when barge-in is temporarily disabled, it is useful to play a unique sound in the background.
System busy audio	When the application is busy processing or fetching contents.
Logo	Unique tones can be used for branding purposes or as auditory cues to identify with an application or portion of the application.
Confirmation tone	Short tones to convey to the user that the input is valid and has been accepted.
Error tone	In case of errors.
Help tone	Unique tone used when about to present help messages (global and local help could also be differentiated by unique tones).
Secure transaction	Unique tones to indicate secure transactions similar to the lock icon in a browser.

It is also important to let callers know specifically that they are talking to a computer and not a real person. Huang [Huang 01] gives a pertinent example, which was used in AT&T’s customer service application. Table 2 illustrates two versions of the opening prompt used in the system.

Table 2. Original and revised prompts used by AT&T’s customer service application [Eisenzopf 14]

Original Prompt	Revised prompt
AT&T Automated Customer Service. How may I help you?	AT&T Automated Customer Service. This service listens to your speech and sends your call to the appropriate operator. How may I help you?

The revised prompt resulted in shorter and more accurate utterances from callers. The original version did not help shorten the utterances, because people did not seem to catch the automated connotation.

With a recognition error rate of about 5% [Huang 01] it is quite common for a speech recognition engine to return the wrong match for an utterance. It is important therefore to confirm what was recognised. There are several techniques for achieving this. Table 3 illustrates two popular approaches. The first approach is an example of confirming, immediately after recognition. This technique is however time-consuming and unnatural from a conversational perspective [Eisenzopf 14].

Table 3. Two caller confirmation techniques [Eisenzopf 14]

Confirmation Technique 1
Computer: What is your first name? Caller: John Computer: So your name is John. Is that correct? Caller: Yes
Confirmation Technique 2
Computer: What is your first name? Caller: Martin Computer: Ok Mary, how old are you? Caller: No, my name is Martin Computer: Oh, sorry Martin. How old are you? Caller: Thirty.

The second approach is similar to how humans interact. This approach includes an implicit confirmation in the next prompt. The recogniser must be programmed to listen for negative responses, such as “no” or “that’s not right”, which tells the system that the name recognised was incorrect. The negative response may be followed by the correct utterance, which should be recognised and reconfirmed [Eisenzopf 14].

Finally, the HFES recommends that unless personnel or cost considerations make it infeasible, callers should be able to access a human representative at any point in the application, should they desire [HFES 01]. This can eliminate caller frustration if the system is having repeated problems recognising the speech of a user.

3. Efficiency or usability factors

Efficiency or usability factors ensure that the task a caller is trying to perform can be accomplished efficiently and that the system is broadly useable, beyond the basic necessity of being purely functional.

The major difference between GUI design and spoken language interface design is that speech recognition can never be perfect. The 5% error rate in speech recognition over the telephone, introduces challenges in creating an effective interface. Consider the difficulty of trying to design a keyboard and mouse interface, which has an error rate of 5% for the key and mouse presses.

Zue and Glass [Zue 01] write that when their system does not recognise a word, users will sometimes try to help by spelling the word, or emphasising the syllables in the word, which usually leads to worse results. This is known as hyper-articulation. Often callers vary their speaking patterns in noisy or stressful situations. For example, they may speak louder their voices may increase in pitch. This phenomenon is known as the *Lombard effect*. This leads to further recognition errors as the system tries to understand the modified speech of a user [Chi 96].

Errors can also be due to any number of different phenomena, including acoustics, speaking style, disfluencies (e.g. um, err), out-of–vocabulary words or understanding gaps. Lombard speech can lead to a “rejection death spiral” in the application [Zue 01]. Callers’ speech patterns become increasingly distorted in order to get the system to understand what they say, which makes users even less likely to be understood.

Systems can respond to errors by giving callers increasingly detailed help messages. For example, in the Jupiter weather system the first rejection generates the message “I’m sorry I didn’t understand you”. Subsequently the system will respond with help messages, intended to encourage the user to speak within the domain [Zue 00b]. Table 4 illustrates how errors can sometimes be avoided by improving the prompts a system might generate [Huang 01].

Table 4. Decreasing error rates by improving dialogue design [Huang 01]

Original Dialogue
Computer: Please speak the name User: Mike Computer: Please speak the name User: Mike
Redesigned Dialogue
Computer: Please speak the first and last name: User: Mike Miller Computer: Which division is Mike Miller working? User: Research group

The “Star Trek Model” of speech interfaces is based on the futuristic communications systems used in the Star Trek films and television programs. In Star Trek, the artificially intelligent devices are capable of verbal interaction comparable to that of a human. Users often believe that the deployment of speech systems will bring them into the realm of science fiction and this can lead to frustration when a speech system fails to perform as expected [Markowitz 96]. The primary reason users turn to the Star Trek model is that it is the only model for speech recognition they have. A user can sit at an unfamiliar PC with an unfamiliar operating system and intuitively know how to use it because they are familiar with the Windows, Icons, Mouse, Pointer (WIMP) paradigm for graphical user interfaces. There is no equivalent for when users are presented with a speech user interface however [Shneiderman 00] [Markowitz 96] [Glass 99]. Callers often do not know how to respond to a speech enabled system. Although this will change as applications become more ubiquitous and standards for interaction emerge [ETSI 02], addressing unrealistic user expectations is a vital facet of planning an application.

User expectations can be addressed by communicating the capabilities of the system to callers as they use the system. This is often achieved by prompting the user in the manner of "Hello, I can give you news sports or weather. Which would you like?" [Glass 99].

Related to the issue of caller expectations is the issue of establishing caller context early in the interaction. Attwater describes two important dimensions to this [Attwater 04]:

Victim or volunteer - Was the caller expecting automation or were they unsuspecting victims?

Frequent or infrequent - Is the caller well primed and experienced or do they rarely call the service.

Establishing these criteria will help establish the user's expectation for the system and consequently the system can prompt the user appropriately.

It is also important to understand the motivation of users calling a voice enabled web system. Systems where user motivation is high are more likely to be successful. Patterson [Patterson 02] has identified caller motivation as one of five caller centric, critical success factors in building voice enabled web systems. An understanding of caller motivation can be used to optimise the system for the tasks which callers are most interested in completing and the information they most need to gain access to. Sharma & Kunins [Sharma 02] recommend gathering user information through the conducting and analysis of one-on-one of group, user meetings, so that relevant data on caller motivation can be established, directly from users.

In telephone based systems it is desirable to let users interrupt the system output at any time, in particular if the output is based on an erroneous understanding or contains superfluous information. This is known as "barge-in". Enabling barge-in can significantly enhance the user experience. Supporting barge-in however presents its own set of challenges, not least of which is detecting a true user barge-in from background sounds like coughs [Ström 00].

4. User satisfaction or desirability factors

User satisfaction or desirability factors determine how comfortable users will be interacting with a voice enabled web system. Although it may be possible to complete a task through the system, addressing these issues will increase the likelihood of a caller using the system repeatedly.

The issue of naturalness in speech systems is one of the most important in advancing user acceptance. Callers are much more likely to interact with a system they feel comfortable with and that responds in a human like way [Markowitz 96]. The ultimate voice enabled web systems would pass the "Turing Test". As Turing put it:

> "*…it is proposed that a machine may be deemed intelligent, if it can act in such a manner that a human cannot distinguish the machine from another human merely by asking questions via a mechanical link…*" [Turing 50]

Limitations of speech recognition technology and machine intelligence mean that this goal is not yet achievable.

Research suggests that people display similar behaviour in interacting with computer systems as they do when interacting with real people [Reeves 99]. It is suggested that because the human brain evolved in an environment where *only* humans exhibited social behaviour, humans tend to respond to objects which exhibit social characteristics as human. This is an evolved response, which can be overcome only when people are consciously aware of their behaviour and choose to reject it. This response explains why, for example, people feel fear when watching a frightening film, even though it is not real. Reeves and Nass [Reeves 99] detail a number of experiments carried out to validate this theory. These experiments suggest that people exhibit characteristics such as politeness, interpersonal distance, flattery, judgement and prejudice when interacting with even the simplest user interfaces. They therefore propose that the design of user interfaces to computer systems should be based on social principles. Human beings are natural experts in social interaction and will respond to computer systems that leverage this skill in humans. Reeves and Nass [Reeves 99] suggest that that people will automatically become experts in interacting with systems, if interfaces to those systems mimic social interactions.

Most research on speech user interfaces suggests creating a consistent personality for the automated voice of the system [Halpern 01] [Eisenzopf 14]. This helps users to relate to the system and feel comfortable in using it. Kotelly [Kotelly 02] proposes that the personality of a voice enabled web system is conveyed in three ways:

- The text of the prompts
- The voice speaking the prompts
- The directing of the prompts

Eisenzopf [Eisenzopf 14] suggests using anthropomorphisms in certain types of applications. Anthropomorphisms are dialogue characteristics that present the system as human. For example, the system should refer to itself in the first person as in "Sorry, I did not understand. Please repeat your request".

The personality presented by the application needs to reflect the brand and image of the organisation. Bullmore proposes that a brand consists of people's perceptions. He writes:

> *"…Products are made by companies. Brands on the other hand are made and owned by people… a brand is a subjective thing…no two people however similar hold precisely the same view of the same brand…"* [Bullmore 01]

He concludes that effective brand communication involves processes which are uncontrolled, disordered, abstract and intuitive. Heath [Heath 01] advances this concept, by proposing that brand information is not actively sought, but passively acquired through an automatic mental process called "low level involvement processing". Low-level involvement processing is a mixture of conscious and subconscious activity. Much of it involves implicit learning. Perceptions and simple concepts repeatedly and implicitly reinforced at low levels of attention tend to define brands in consumers' long-term memory. Because implicit memory is more durable, these brand associations once learned are rarely forgotten.

Fig 1. presents a taxonomy of factors that can communicate brand information in a voice enabled web system. The brand is reinforced through dialogue cues such as the personality of the speaking voice, prompts played and grammar understood. The system might use informal language and colloquial speech in the prompts, if appropriate to the brand. Non-dialogue audio cues, such as background music, earcons and the stating of the tagline of the company can also communicate brand information. More subtle indicators such as the reliability of the system can help communicate the message of a reliable and trustworthy brand. It can be argued that people draw conclusions about the underlying competence of computer applications in a way that is similar to how people draw conclusions about humans. If the system apologises too much when it makes an unimportant error, or if it talks too slowly, it may create the impression that the system is incompetent. Having a dialogue that creates the impression of an enthusiastic competent helper is important in inspiring confidence in users [Sharma 02].

Lawrie [Lawrie 02] presents the example of "Julie", the voice of US rail company, Amtrac. Amtrac opted for a casual, conversational approach to the interface of its voice enabled web timetabling and ticket booking system. Julie greets all callers in a warm, friendly manner and provides regular reassurance as she navigates callers through the speech service. Since speech enabling the service, automation rates have increased by 61%.

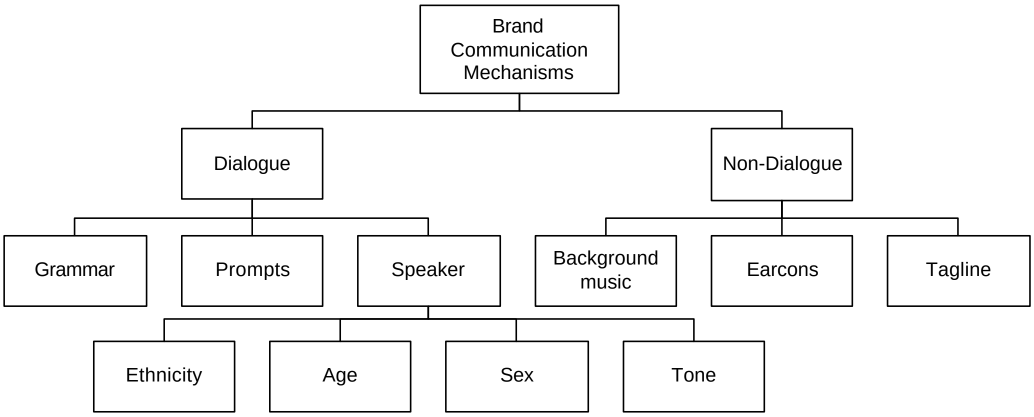

Fig 1. A taxonomy of factors that communicate brand information in a voice enabled web system

The usage of text to speech is another important issue in creating a natural sounding interface. Available literature would suggest that natural sounding text to speech is not yet achievable using technologies available today [Dutoit 97] [Zue 01] [Economist 01]. One popular approach to solving this problem is to concatenate pre-recorded speech units together to form an utterance. Text to speech is then used in parts of the system where this is not possible, for example in reading emails [Duggan 03a].

With the goal of building speech interfaces that are as natural as possible, speech systems designers should examine human-human interactions.

Human dialogue contains disfluencies, interruptions, confirmation, clarification and sentence fragments. Consider the transcript from a conversation in Table 5 for examples of these [Zue 01].

Table 5. Transcript of a conversation between an agent A and a client C. Typical conversational phenomina are annotated right [Zue 01]

Dialogue	**Speech Characteristic**
C: Yeah, [umm] I'm looking for the Buford Cinema.	Disfluency
A: OK, and you want to know what's showing there or…	Interruption
C: Yes, please	Confirmation
A: Are you looking for a particular movie?	
C: [umm] What's showing?	Clarification
A: OK, one moment	Back-channel
C: They're showing a Troll in Central Park	
A: And the others?	Fragment
C: Little Giant	Fragment
A: That's it	
C: Thank you	
A: Thanks for calling Movies Now	

As completely human-like conversations are difficult to build, many systems incorporate mixed-initiative, goal directed dialogue, where both the user and the computer participate interactively using a conversational paradigm. In these systems it is often necessary to ask users to interact with the system in a way that is more structured. System should possess some of the characteristics of a human agent however, so that users can feel comfortable using them [Zue 01].

Users may not always repeat commands as explicitly requested of them in an application. For example if an application prompted a caller with “Would you like film or theatre listings?” callers could answer with any of the following “theatre, give me theatre listings, I would like theatre listings, erm, theatre please” and so on. It is therefore important for the application to support grammars based on real world interactions with the system. It is important for developers to study utterances collected from application logs to add new phrases as well as delete unused phrases.

Silence on behalf of a user should be interpreted as requiring attention. This may be the result of a user not understanding a prompt. Any pause over 2.25 seconds be responded to by the system [Sharma 02]. A response can consist of replaying the prompt, providing further detailed re-prompting, or replaying the help message.

As a general rule, the system should avoid silence. Silence during a phone call often leads users to believe that there is a problem. Many will hang up and others will try to talk to the system out of turn. People's tolerance for “dead air” is limited to two to three seconds, so reducing latency is crucial [Eisenzopf 14].

There are situations where a short pause is appropriate however. Consider the example in Table 6. The system needs to place a short pause, possibly with a busy earcon, between its two prompts. If there is no pause between the prompts, callers may wonder why they were asked to hold on in the first place.

Table 6. The importance of using pauses [Sharma 02]

Caller:	What is my account balance?
Computer:	Hold one for a moment while I check
Computer:	Sorry there is a problem

5. Evaluating a voice enabled web project

The Review Achievement Framework (RAF) is a tool to analyse voice enabled web projects [Duggan 03b]. Each section of RAF consists of a series of questions, which should be answered on a scale of *–2* to *+2*. *–2* indicates a strongly negative response to the question. *–1* indicates a negative response. *0* indicates a neutral response to the question. *1* indicates a positive response and *2* indicates a strongly positive response.

Each response is multiplied by a question specific weighting factor. Weights applied should be decided in advance by the project management team. Different project types will weight responses differently, based on the response’s perceived impact on the overall success of the project. Branding for example, might have a lower weighting than the preparation of test plans for an internally targeted system. If the system were to be used by external customers however, then branding might be rated higher. The entire framework contains of 49 questions covering legal, financial, HCI, planning, organisational, technical and personnel issues. The framework was applied to a case study supplied by VoxPilot [VoxPilot 04] and found to be a useful indicator of weaknesses in the lifecycle of the project. VoxPilot is a company providing outsourced solutions for the development, integration and deployment of VoiceXML applications for the European market.

Table 7 presents an adapted version of RAF, consisting specifically of the issues related to HCI outlined in this paper. It is proposed that this table might be a useful checklist for developers implementing voice enabled web systems in order to verify that relevant issues have been addressed. Weighting columns have been removed for the sake of simplicity.

Table 7. RAF-HCI: Questions to establish how well HCI issues in voice enabled web systems have been addressed [Duggan 03b]

RAF – HCI

	Question	**Response**
1	Has user profiling been carried out on the target user demographic of the system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
2	Does the system cater for non-native speakers?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
3	Are earcons used to provide context?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
4	Does the system indicate in its opening prompt that it is a computer and not a real person?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
5	Does the system use a confirmation technique to improve accuracy?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
6	Is a transfer to a human agent possible, if callers so desire or if speech recognition fails?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
7	Has hyper-articulation and Lombard speech been modelled in the speech recognition sub-system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
8	Are increasingly detailed prompts given in response to recognition errors?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
9	Does the system address caller expectations by communicating the capabilities of the system to callers as they use the system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
10	Does the system support differing levels of prompts, based on user context (frequent or infrequent, victim or volunteer)?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
11	Are incentives provided to motivate callers to interact with the voice-enabled system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
12	Does the system support barge - in?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
13	Does the system use anthropomorphisms in prompts?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
14	Does the system have an identifiable personality?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
15	Do trained actors professionally record all prompts (I.e. is there a minimal use of text to speech?)	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
16	Are flexible, human modelled grammars supported?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
17	Does the system respond appropriately to user silence (e.g. with re-prompting)?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
18	How effectively has the organisation's brand been incorporated into the system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
19	Are busy earcons used to cover any pauses in the system?	-2 -1 0 1 2 ☹ □ □ □ □ □ ☺
	Total:	

6. A case study

"Red Nose" day is a national charity event organised by the BBC and ITV television stations in the UK. For Red Nose day 2003, VoxPilot [VoxPilot 04] developed a voice driven telephone game for their client Saffron Interactive. Saffron Interactive is an online training company who provided the funding for the development and hosting of the game, with all proceeds going to the Red Nose charity.

The game could be accessed via a premium rate telephone number. Callers were charged 60p (Sterling) per minute to call the game, with 40p per minute going to charity. The game consisted of rounds of random multiple-choice questions. The top prize callers could win was a holiday. Callers had the option to leave their personal details, which the system recorded. Callers could call the system to play the game as often as they liked. Each time they called the system they could find out where they scored relative to other players as the system maintained a league table of the top players. This provided a motivation for people to call the system often as they could check their positions in the league table and improve their scores by answering more questions.

VoxPilot regarded the system as relatively small. They had developed similar applications in the past and they leveraged their skills, tools and existing code to propose an aggressive timeframe for the project. VoxPilot undertook the development and hosting of the application on behalf of their client and turned around the project, from initial consultations through to deployment in 4 weeks. VoxPilot provided a full turnkey outsourced solution for Saffron Interactive.

Due to the aggressive timeframe of the project, extensive prototyping was not undertaken, however a single prototype was developed using text to speech (TTS). In the final version of the application however, all prompts were professionally recorded by experienced voice actors. The delivered application used simple directed dialogues so as to minimise grammar complexity and reduce error rates.

From a technical standpoint, VoxPilot and their client regarded the project as a success. The final version of the application exceeded the original specifications and was delivered on time and within budget. It incorporated an innovative caller motivation system (the league table), context earcons and an identifiable personality. From a caller perspective, the system was also regarded as successful. The system suffered low recognition error rates and was compelling to use. This was accounted for by the simple dialogue styles adopted and by the high standard of voice user interface (VUI) design expertise of the team developing the system.

From a commercial standpoint however, the project did not deliver the expected return on investment. This was accounted for by the unsuccessful marketing strategy employed. There is a direct correlation between the amount of money invested in marketing a voice enabled web system of this type and the return the system will generate. Saffron Interactive began their marketing of the system on Red Nose day itself and followed this up with additional advertising several days after the event. It was felt that marketing of the system did not incorporate the required synergy with the Red Nose branding for the system to achieve the necessary traction in the marketplace.

Applying RAF to the overall project highlighted a number of successful areas and also identified weaknesses in the project lifecycle. The use of the case study additionally highlighted areas where RAF could be evolved [Duggan 03b]. Some of the key findings related to interface development are outlined below:

- The target demographic for the application was broad and the grammars were simple, so user profiling was not carried out.
- No special consideration was given to non-native users of the system, although some grammar modifications were made post-implementation, based on live dialogue transcripts. This helped increase recognition accuracy.
- The system incorporated various earcons, which not only provided context, but also enhanced to overall caller experience.
- The importance of the system identifying early on in the interaction that it was using speech recognition was emphasised.
- As grammars employed in the system were simple, a confirmation technique was not employed.

- Transfer to a human agent was not implemented as it was not appropriate given the type of application being delivered.
- The importance of providing increasingly detailed prompts in response to errors was emphasised.
- Hyper-articulation and Lombard speech patterns were not explicitly modelled in the application though the recognition engine used was sufficiently sophisticated to be able to recognise speech in the majority of these cases.
- The importance of providing increasingly detailed prompts in response to recognition errors was acknowledged.
- It was not deemed necessary to explicitly support different levels of user, given the short projected lifespan of the application.
- There was a high motivation to call the system. There was a possibility of winning a holiday and helping a charitable cause. The league tables were felt to provide an additional incentive for repeat calling.
- The system supported barge-in where appropriate. This was felt to be essential, given that a caller might know the answer to a question and would not appreciate having to listen to all the alternatives.
- The system used anthropomorphisms in prompts and had an identifiable personality. It was felt that that this is particularly important for consumer-facing systems.
- Although the prototype system used TTS, all prompts in the released version were professionally recorded.
- The commercial failure of the system was attributed to 2 issues. Firstly was the lack of up-front consideration given to marketing the finished system. Secondly and related to the first issue was the fact the system failed to capitalise on the “Red Nose” brand.

7. Conclusions

This paper outlined a number of major HCI considerations relevant to building speech based telephony user interfaces. The paper categorised these issues according commonly used criteria for building effective, efficient and desirable interfaces. It presented a simple developer checklist for ensuring that HCI issues have been addressed. It also presented a case study of a commercial application which was analysed by the criteria presented in the checklist.

To create an effective and useful interface, user profiling should be carried out to ensure that callers speaking patterns are correctly modelled. To ensure that callers are aware of the context of the interaction, the interface should implement user feedback and confirmation. The system should manage caller expectations by using directed prompts and access to a human operator should be available to callers where speech recognition completely fails.

To create an efficient and usable interface, the system should incorporate error handling techniques, such as playing increasingly detailed prompts in response to the inevitable recognition errors. The system should provide short cuts for frequent callers and support *barge-in* so that callers can interrupt the computer.

To create a desirable user interface that leads to high levels of caller satisfaction, the application should endeavour to create a natural sounding interaction with the caller. The system should refer to itself in the first person. The speaking voice and non speech audio cues in the application should create a positive association with the branding of the organisation. Human dialog phenomena such as disfluencies, interruptions, confirmation, clarification and sentence fragments should be modelled in the dialogues.

Caller silence on behalf of a user should be interpreted as requiring attention, and the system should respond appropriately. In general, the system should avoid silence and instead play a busy earcon while it is processing.

Given advances in speech recognition and text to speech technology and the maturing of W3C standards for speech interfaces, it is understandable why deployments of voice enabled web technology are growing. Increasingly, routine calls to call centres are being answered by computers that use speech to communicate with callers. As is the case with graphical interfaces, the use of speech as a user interface modality should be governed by established guidelines and best practice, such as those outlined in this paper.

REFERENCES

[W3C 04] W3C.: Frequently asked questions, http://www.w3.org/Voice/#faq, Accessed February, 2004

[Fitzpatrick 98] Fitzpatrick, R. and Higgins, C.: Usable software and its attributes: A synthesis of software quality, European Community law and human-computer interaction, In: People and Computers XIII. Proceedings of HCI98 Conference, Springer, London, UK 1998

[Markowitz 96] Markowitz, J.: Using Speech Recognition,, Upper Saddle River, NJ : Prentice Hall, 1996

[HFES 2001] HFES.: HFES ANSI 200: Human Factors Engineering of Software User Interfaces, Human Factors and Ergonomics Society 2001

[Sharma 02] Sharma, C., Kunins, J.: VoiceXML: Strategies and Techniques for Effective Voice Application Development with VoiceXML 2.0, John Wiley & Sons 2002

[Economist 01] Economist , The: Just talk to me, December 6 2001

[Glass 99] Glass, J. R. Challenges for Spoken Dialogue Systems, Proc. 1999 IEEE ASRU Workshop, Keystone, CO, December 1999

[ETSI 02] ETSI ES 202 076 - Human Factors (HF); User Interfaces; Generic spoken command vocabulary 2002 for ICT devices and services

[Yuk 99] Yuk, D., Flanagan, J.: Telephone Speech Recognition using Neural Networks and Hidden Markov Models, Dept. Of Computer Science, Rutgers University, NJ., 1999

[Van 01] Van Compernolle, D.: Recognising Speech of Goats, Wolves, Sheep and…Non-Natives, Speech Communication, Vol 35 pp. 71-79, 2001

[Zue 00a] Zue, V., Glass, J.: “Conversational Interfaces: Advances and Challenges”, Proceedings of the IEEE, Vol 88 No 8, August 2000

[Nuance 04] Nuance: Nuance Languages, http://www.nuance.com/prodserv/availablelanguages.html, Accessed February 2004

[Horton 94] Horton,W.: The Icon Book, New York, John Weilly, 1994

[Huang 01] Huang, X.D., Acero, A., Hon, HW., Reddy, R.: Spoken Language Processing: A Guide to Theory, Algorithm and System Development, Prentice Hall, 2001

[Eisenzopf 14] Eisenzopf, J.: Top 10 Best Practices for Voice User Interface Design http://www.developer.com/voice/article.php/1567051, Accessed April 2003

[Chi 96] Chi, S & Oh, Y.: Lombard Effect Compensation And Noise Suppression For Noisy Lombard Speech Recognition, The Fourth International Conference on Spoken Language Processing, Philadelphia, USA, December 1996

[Zue 00b] Zue, V., Seneff, S., Glass, J., Polifroni, J., Pao, C.: Jupiter: A Telephone-Based Conversational Interface for Weather Information, IEEE Transactions on Speech and Audio Processing, January 2000

[Shneiderman 00] Shneiderman, B.: The limits of speech recognition, Communications of the ACM, Volume 43 , Issue 9 September 2000

[Attwater 04] Attwater, D., Edgington, M.: Oasis - A Framework for Spoken Language Callsteering, BT Labs

[Patterson 02] Patterson, S.: Global Speech Day, Keynote Speech, Global Speech Day, May 2002

[Ström 00] Ström, N. & Seneff, S.: Intelligent Barge-in in Conversational Systems" Proc. 6th International Conference on Spoken Language Processing, Beijing, China October 2000

[Turing 50] Turing, A.M.: Computing Machinery and Intelligence, Mind, 1950

[Reeves 99] Reeves, B., Nass, C.: The Media Equation : How People Treat Computers, Television, and New Media like Real People and Places, C S L I Publications, 1999

[Halpern 01] Halpern, E.: Human Factors and Voice Applications, VoiceXML Review, June 2001

[Kotelly 02] Kotelly,B.: The Science Behind Successful Caller-Experience, Global Speech Day presentation, May 2002

[Lawrie 02] Lawrie, C.: Best Practices: Achieving Success with Speech, Speech Technology Magazine, November/December 2002

[Dutoit 97] Dutoit, T.: Text, Speech And Language Technology, Kluwer Academic Publishers, Dordrecht, April 1997

[Duggan 03a] Duggan, B., Deegan, M.: Considerations In The Usage Of Text To Speech (TTS) In The Creation Of Natural Sounding Voice Enabled Web Systems, International Symposium on Information and Communication Technologies, Trinity College, Dublin 2003

[Bullmore 01] Bullmore, J.: Posh Spice & Persil, British Brands Group Annual Lecture, 2001

[Heath 01] Heath, R.: The Hidden Power of Advertising, World Advertising Research Center; August 2001

[ISO 98] ISO.: International Standardisation Organisation (ISO): “ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability” http://www.iso.org, 1998

[Duggan 03b] Duggan, B.: Strategies for Enterprise Voice Enabled Web Projects, MSc Dissertation, School of Computing, DIT, 2003 http://www.comp.dit.it/bduggan/research.htm

[VoxPilot 04] VoxPilot: http://www.voxpilot.com/, Accessed May, 2004