

Voice User Interface Design for a Telephone Application Using VoiceXML

Daniel Mekanovic and Hao Shi

School of Computer Science and Mathematics
Victoria University, Australia
hao.shi@vu.edu.au

Abstract. VoiceXML is a standard language for developing voice based applications. VoiceXML applications have more advantages over traditional Interactive Voice Response (IVR) systems because they can be used through any type of phones and also accessed via a computer. Voice User Interface (VUI) design is an integral part of developing any VoiceXML application. In this paper, the VUI for a VoiceXML ‘Cinema Service’ telephone application is designed and a number of experiments are undertaken to help the design of the VUI. The experiments focus on users’ navigation, memory and age group, and preferences. Conclusions are drawn based on the experiments for future design and development.

1 Introduction

Most businesses today provide customer services/help via the web and telephone. Live-operator call centers can not handle large amounts of simultaneous calls. This often leads to long call queue/waiting and frustrated customers.

Applications with current voice technologies such as Interactive Voice Response (IVR) are run on propriety platforms, require specialist skills to develop and maintain, and are not portable or flexible. Barge-in, security, and voice recording features are rarely found in IVR systems.

VoiceXML allows a developer with basic programming knowledge to create a fully functional, customizable and dynamic voice-based application. Voice User Interface (VUI) design is an integral part of developing any VoiceXML telephone application just like a Graphical User Interface (GUI) is very important when designing a visual application.

In this paper, a VUI is designed for a ‘Cinema Service’ telephone application using VoiceXML. A series of experiments are conducted in order to understand user experience, improve the initial VUI design and provide valuable information for the future VUI development.

2 VoiceXML and Voice User Interface (VUI)

VoiceXML is a W3C standard mark-up language for scripting voice interactions between a computer and a person. It is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of spoken and DTMF (Dual Tone Multi-Frequency) key input, recording of spoken input, telephony, and mixed initiative conversations. VoiceXML uses natural dialog. Its major goal is to bring the advantages of Web-based development and content delivery to interactive voice response applications [1]. Since the technology uses XML syntax, it can be easily used with other XML based technologies, such as Web Services, to send and receive data/information via the Internet.

Designing a VUI is very different from designing other user interfaces. Most user interfaces usually offer some visual information for users to interact with. With VUI, users have to listen, memorize, and speak to interact with the system. The VUI cannot provide rich content information because users are not able to memorize pages of verbal information. The design has to be simple, with short and clear dialogs so that users can navigate through the service without any problem. Speech recognition technology is still imperfect, and users encounter failure for various reasons. It is important to accommodate errors when designing a VUI. Also users possess a wide variety of voice, speech skills and vocabularies, and all these factors must be considered.

3 VUI Design for a Telephone Application

A VUI for a 'Cinema Service' telephone application is designed using Jakob Nielsen Ten Usability Heuristics [2] and hosted by a voice hosting service provider. The application provides the following functions:

- Search for movie by name, genre, classification, release date and actor
- Listen to movie reviews, ratings and trailer
- Find session times

The following is a sample VoiceXML code as part of VUI design for the telephone application:

```
<?xml version="1.0" ?>
<vxml version="2.0">
<form id="greeting"><block>Welcome to the Cinema service.
<goto next="#main"/></block>
</form>
<menu accept="approximate" dtmf = "false" id = "main">
<prompt>You can book movie tickets, find session times, ...
What would you like to do?</prompt>
</menu>
<form id="book">
<block> Booking movie tickets. <goto next = "book.vxml" />
</block>
</form>
...
</vxml>
```

4 Experiments

Designing a VUI is a relatively new experience for most developers. There are very few good VUI guidelines available today mainly because VUI design is so new and requires ample usability testing [3]. In order to improve the initial VUI design, a series of experiments are conducted on the ‘Cinema Service’ telephone application.

4.1 Experiment 1 – Navigation

The aim of this experiment is to find out the experience of users in navigating through the application. Users are not given any instructions.

During the experiment, users perform the following sequence of tasks:

Task 1: Find out the running time of the movie “Superman 5”.

Task 2: Find the five-star rating of the movie “Matrix 4”.

Task 3: Find all session times for “Aliens vs. Predator” on Friday night.

Task 4: Book two movie tickets for “Predator 3” at noon on Tuesday.

Task 5: Find all movies starring “Tom Jones”.

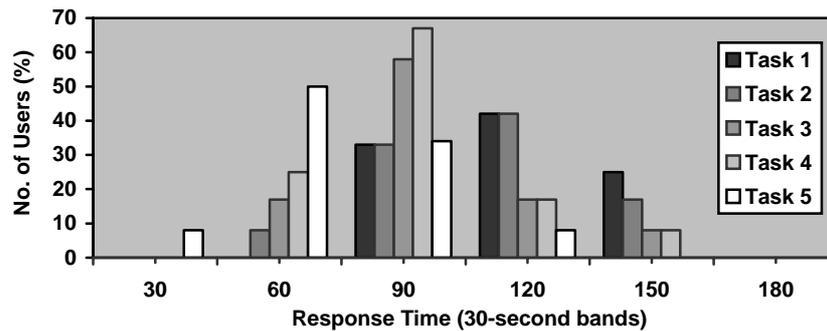


Fig. 1. Results of Experiment 1

The experimental results in Fig.1 show that all the users are slow at the beginning and become more comfortable with the application and improve their skills progressively. In the end users start to predict and ‘barge in’. It indicates that an advanced user interface should be created for experienced users.

4.2 Experiment 2 – Memory and Age Group

The second experiment aims to find out how long and how many voice prompts can be spoken to a user before the user forgets previous prompts or gets confused.

Four separate menus are created for this experiment.

Menu 1: Four links, each link has a brief description

Menu 2: Four links, each link has a long description

Menu 3: Eight links, each link has a short description

Menu 4: Eight links, each link has a long detailed description

Users find all the menus easy to remember, except for the last menu which is long and detailed. However, the experimental results show that there is no correlation between age groups (18 to 39 and 40 plus) and the number of links that can be memorized.

4.3 Experiment 3 – TTS Settings and Pre-recorded Prompts

This experiment intends to investigate user-preferred application settings. Users are asked to answer the following questions:

- Do you prefer a male or female voice?
- How do you understand the TTS (Text-to-Speech) voice?
- Do you prefer TTS or pre-recorded human voice prompts?

Table 1. Results of Experiment 3.

Gender		Understanding TTS		Voice Preference	
Female	75%	All	75%	TTS (Computer)	33%
Male	25%	Most	25%	Pre-recorded (Human)	67%

As indicated in Table 1, majority of users prefer female TTS voice. Nearly all users can understand TTS voices. And pre-recorded human voice is preferred over the computer generated TTS voice. Due to the fact that all the dynamic contents have to be played to users via the TTS, female TTS voice is the choice.

5 Conclusions

A VUI for a VoiceXML telephone application has been designed and a number of experiments carried out. From the experimental results, it can be concluded that two separate user interfaces should be created so that users with different levels of experience can choose a different interface. Multiple, long and descriptive prompts make navigation difficult and female TTS voice is preferred for dynamic VUI.

References

1. McGlashan, S. et al.: Voice Extensible Markup Language (VoiceXML) Version 2.0, <http://www.w3.org/TR/voicexml20/>, W3C Recommendation 16 March 2004.
2. Nielsen, J.: Heuristic Evaluation of User Interfaces, Proc. of ACM CHI'90 Conference. Seattle, WA, 1-5 April (1990) 249-256.
3. Yankelovich, N.: Designing Effective Speech Interfaces, John Wiley & Sons, Inc. (2000)