Designing an Enhanced VUI Tree Framework for e-Education System

Ambrose A. Azeta, Christopher K. Ayo, Aaron A. Atayero, Nicholas A. Ikhu-Omoregbe

College of Science and Technology, Covenant University, Ota, Ogun state, Nigeria

azeta_ambrose@yahoo.com, ckayome@yahoo.com, atayero@ieee.org, nomoregbe@gmail.com

Abstract. VoiceXML is a standard language recognized by World Wide Web Consortium (W3C) for developing Voice User Interface (VUI) applications. Advances in speech recognition and VUI systems have motivated the provision of a VUI design framework for interacting with vocal applications. For most developers, designing a VUI is a relatively new experience. Today, good VUI frameworks are few mainly because VUI design is novel and requires usability criteria for users' friendliness and ease of navigation during interaction. This paper presents an enhanced VUI tree framework for Voice-Enabled E-Education System (VEES) using Multi-RootNode Tree to improve speed of access and user's flexibility while interacting with the system. The propose system will be of immense benefit to the normal users and the visually impaired learners since it will make navigation, access speed and interaction process with the system a lot easier.

Keywords: e-Education; Framework; VEES; VUI Tree; VoiceXML

1. Introduction

In the past decade, the Internet has offered learning institutions a new medium for efficient sharing of learning resources. This has made educational institutions to continue investing in e-Education business for efficient delivery of Learning Management System (LMS) using the World Wide Web. Despite the Internet's growing acceptance, the telephone network is still more widely and readily accessible [1]. The Internet is increasingly becoming accessible through telephone (voice). Voice technology, and, especially VoiceXML, is the enabling technology that allows people to interact with computers and the Internet over the telephone using natural speech dialogues. VoiceXML technology allows people to access the web from anywhere they can make a call, thus providing increased access to goods and services, as well as to customers [2]. Telephone is simple to operate and is the most natural form of communication, i.e. it uses the human voice. The proliferation of mobile devices in the last decade has made access to the telephone network even easier. As discussed in [3], every successful voice-enabled application is built on three solid foundations: Firstly, universal user interfaces guidelines and usability principles. Secondly, the project guidelines for the application and system requirements. Thirdly, the interaction guidelines for the personality that will be portrayed by the Voice User Interface (VUI). This foundation requires that a standard VUI framework be followed as a guideline for building a voice interface application.

Providing a voice-based interface with the web interface is an advantage to the visually challenged who are unable to use a visual interface. It is also possible to use the application for accessing a web-based interface even while on the move through a mobile phone, which is much easier to carry around than a personal computer. Phone applications are more spontaneous. Most people these days always have their phone on their hip. In many cases, the phone transaction can be completed before the PC even boots or you can log in. The number of telephone users is far greater than the number of people that uses personal computers or the Internet.

In voice-enabled telephony systems, users interact with machines in a conversational dialog over a phone connection. The design of such dialogs affects the quality of service offered by the system. Hence, Voice User Interfaces (VUIs) should be carefully designed with high quality interactive management techniques to attain a positive usability attribute particularly in the area of user's satisfaction, ease of navigation and accessibility. One of the most significant elements of a good user interface is visibility of the system's status. In an ideal situation "What you see is what you get" (WYSIWYG), but with telephony system, "What you hear is what you get" (WYHIWYG). That is, if software VUI interface hears the word *exam* and *exam* is understood, then *exam* happens. Users must also notice immediately what's going on behind the scenes and whether their actions have actually led to the expected results [4].

Voice-enabled applications are currently being applied in e-Education [5, 6, 7, 8]. Thus, by using VoiceXML applications, we can reach out to more students than is possible by using the Internet. This is so because the educational environment is one of the most popular sectors where majority of people carry mobile phones. One of the goals of an e-Education system includes providing quality learning objects to students at a considerable cost saving through a voice-enabled learning platform such as in Fig. 1. This goal can easily be achieved with a voice-enabled application. One way of building quality into technology supported education is providing a successful VUI that is user-friendly and is easily navigated to encourage learners to continue using the e-Education platform. 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.

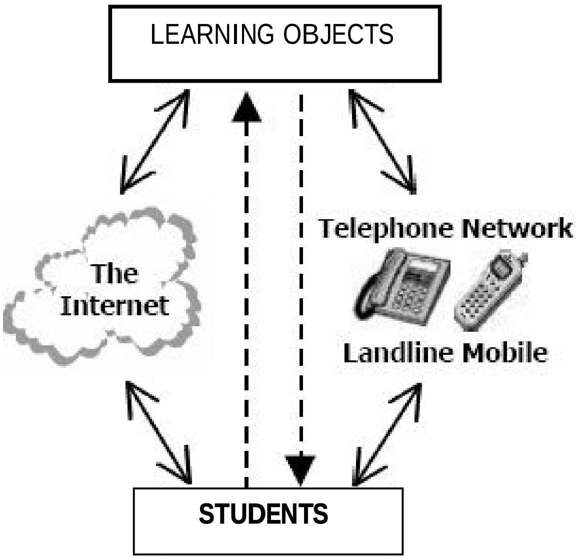

Fig. 1. Voice-enabled Learning platform. Source: Adapted from Phonologies [1]

There are several voice-enabled frameworks in the areas of e-Education [7, 9, 10]. However, there is death of a user friendly VUI framework for interaction management and the speed of accessing voice applications is considerably low. The depth of menu-driven navigation for VUI application has been a problem for certain category of users. We should also consider the fact that 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 [11].

This paper explores the design of a VUI tree framework using an enhanced Multi-RootNode Tree structure to improve the users' mode of interaction. The remaining parts of the paper are structured as follows: Section 2 describes the VUI tree framework and the optimization techniques for improving the framework. Section 3 highlights the proposed enhanced Multi-RootNode Tree technique for restructuring VUI menu-driven applications and section 4 concludes the paper.

2. VUI Tree Framework

The VUI Navigation was modeled using VUI tree structure as shown in Fig. 2. When a user named catman uses a telephone and makes a phone call to *VEES*, s/he will reach the root node of the VUI tree and hear a greeting message, which may look like "Welcome to intelligent voice enabled e-Education Application". If you would like to select student menu say student, for faculty menu say faculty and admin menu say admin. Catman may say faculty and traverse the VUI tree to N2. The interaction between user catman and the system will be confirmed when catman navigates to level two of the VUI tree. At this point he can listen to the registration.

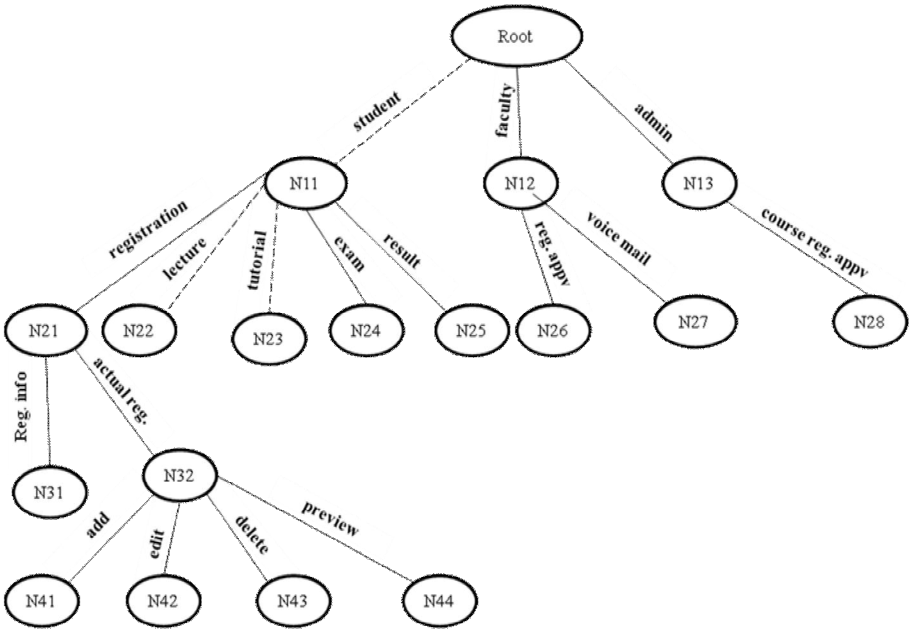

Fig. 2. A VUI tree framework for the system

2.1 VUI Optimization and Enhancement

One approach for optimizing VUI in [12] is the operation of node promotion/demotion (by moving some nodes to higher/lower levels of the VUI tree). This approach uses a Genetic Algorithm based optimization technique. In this model, *root* node is at level 0, and N_{ij} is the *j*th node at level *i*. *Node access time* is the time length of audio heard by users during each visit to the node. *Node access count* is the frequency users navigate through and access to the node in a specific term. *Leaf node access time* is the total access time users navigate from *Root* to the leaf node. *Total time* is the summations of every *leaf node access time* multiply *leaf node access count*. In a voice site, users' destination must end at leaf nodes; therefore, we focus on the leaf nodes in the model. It means only leaf nodes have the property of "access counts."

The properties of the nodes used in this example are described in Table 1. In this case, the values of *node access time* are randomly created in the range between 3 and 8 seconds, the values of *node access count* are only applicable to leaf nodes and randomly generated between 40 and 100, and each value of *leaf node access time* is derived from the values of *node access time* of all nodes on the path from the *root* navigating to that particular leaf node. For example, the *leaf node access time* of *N22* is the summation of *node access time* values of the *root*, N11, N21 and N22 (i.e., 5 + 6 + 5 + 3 = 19), and the *leaf node access time* of *N23* is the summation of *node access time* values of the *root*, N11, N21, N22, N23 and N23 (i.e., 5 + 6 + 5 + 3 + 6 = 25). Using the VUI optimization mechanism, we can restructure the VUI tree and calculate the new total time. The new tree structure is shown in Fig. 3 (nodes having star mark [*] are restructured), and the properties of the nodes of the restructured tree are described in Table 1. Note that while the values of *node access time* and *node access count* on all nodes are unchanged, the values of *leaf node access time* on some leaf nodes are changed because the VUI tree is restructured. In this sample case of optimization model in [12], the improvement is about 57% i.e. (3830-1660=2170, (2170/3830*100) = 57%. Using our enhanced model, the improvement is about 79% i.e. (3830-810=3020, (3020/3830*100) = 79%, even though this is just a simple illustration.

The calculation of total access time for the total access time, optimized time and enhanced time is contained in Table 1. The leaf node access time *(t)* for *N22 has changed from 19 to 8 and for *N23 from 25 to 11 as a result of the optimization. Similarly, the leaf node access time *(t)* for **N22 has changed from 8 to 3 and for **N23 from 11 to 6 as a result of the enhancement. The values were derived using the formula in equation (1), (2) and (3):

$$\textit{Total access time} = \sum_{i=1}^{L} Nij(t) * Nij(c), \quad j = 1..n \tag{1}$$

$$\textit{Optimized access time} = \sum_{i=1}^{L} Ni(t) * Ni(ot), \quad j = 1..n \tag{2}$$

$$\textit{Enhanced access time} = \sum_{=1}^{L} \mathrm{Ni}(t) * \mathrm{Ni}(et), \quad j = 1..n \tag{3}$$

Therefore total time = 19 x 70 + 25 x 100 = 3830, Optimized total time = 8 x 70 + 11 x 100 = 1660, and Enhanced total time = 3 x 70 + 6 x 100 = 810. Several other approaches for optimizing VUI tree include the creation of extra links (for providing short-cuts to VUI tree navigation). Changing the order of child nodes for every nonleaf node can create many different restructured VUI trees, and every VUI tree represents a possible solution. One demerit of node order change is that for a VUI tree with hundreds of nodes, the potential number of restructured VUI trees can be very high.

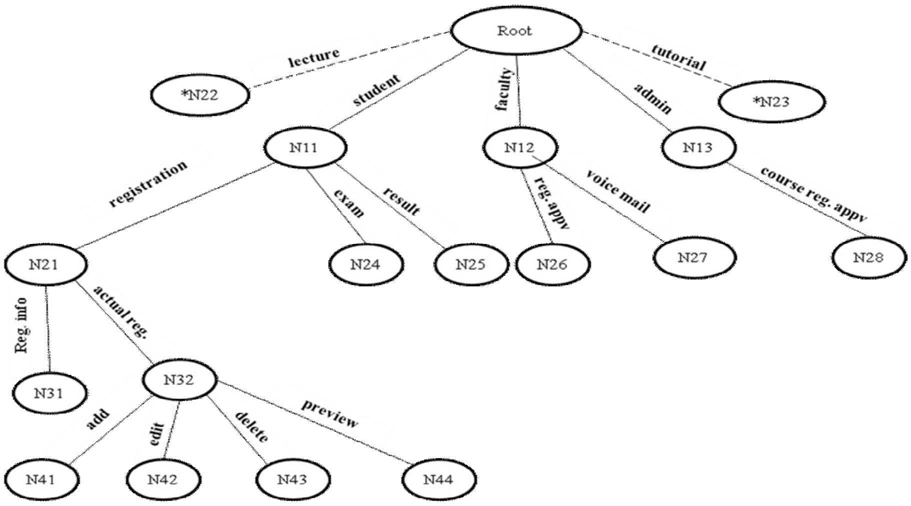

Fig. 3. The restructured VUI tree (Node marked * were promoted)

Table 1. Properties of the nodes

Node	Root	N11	N12	N13	N21	N22	N23	N24	Total Time
Node access time (a)	5	6	5	4	5	3	6	-	-
Leaf node access time (t)	na	na	na	na	na	19	25	-	-
Node access count (c)	na	na	na	na	na	70	100	-	-
Total time	-	-	-	-	-	133(	250(	-	3830
Optimized Leaf node access time (ot)	na	na	na	na	na	8	11	-	-
Optimized total	-	-	-	-	-	560	110(	-	1660
Enhanced Leaf node access time (et)	na	na	na	na	na	3	6	-	-
Enhanced total	-	-	-	-	-	210	600	-	810

2.2 The Enhanced Multi-RootNode Tree Model

With the VUI enhanced Multi-RootNode approach, we can restructure the VUI tree and calculate the new total time as shown in Table 1. The new tree structure is shown in Fig. 4

(nodes having double star mark [**] are restructured). The node label lecture and tutorial have been moved to the same level with the root node.

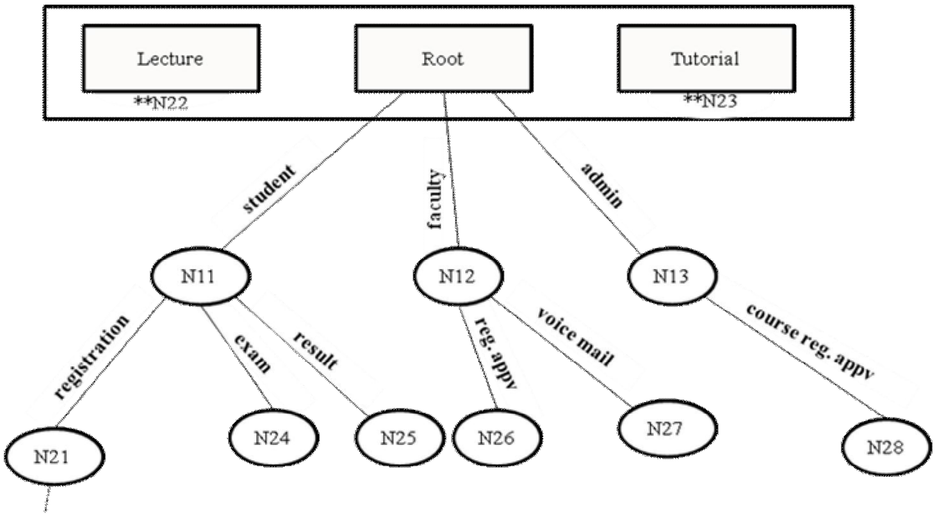

Fig. 4. The VUI tree (Node marked ** were restructured)

2.3 VUI Call flow

Fig. 5 shows a partial Call flow of the proposed system where voice lecture and tutorial menu have been promoted up to be among the root node.

```
VEES: Welcome to intelligent Voice enabled eEducation System
VEES: Voice lecture
VEES: Root
      VEES:  Student menu
      VEES: Faculty menu
      VEES: Admin menu
VEES: Tutorial
--
VEES: Welcome to Voice lecture menu
VEES: Mention the course code to study
VEES: CSC317
--
```

```
VEES: Welcome to student menu.
VEES: registration
VEES: Voice lecture
VEES: tutorial
VEES: examination
VEES: result
VEES: exit
--
```

Fig. 5. Partial Call flow for the system

3. Conclusion

In this paper, an enhanced VUI tree framework for improving the speed and interaction management of voice-based e-Education system (VEES) has been presented. The framework would serve as a reference model for implementing user-friendly telephone-based e-Education applications for normal and visually impaired learners. The application when fully implemented would assist people with physical access difficulties (e.g. repetitive strain injury, arthritis, high spinal injury) that make writing difficult. It can also be effective for students with reading or spelling difficulties (e.g. dyslexia). The VUI tree framework provided in this study will be of immense benefit to the research community.

References

1. Phonologies, "Phonologies (India) Private Limited (2001),. Retrieved 29 May 2010 from http://www.phonologies.com/pdfs/whyvoicexml.pdf
2. Abbott K. R. (2002). "Voice enabling Web applications: VoiceXML and beyond". APress, Springer- Verlag, New York.
3. Tucker D."Voice User Interface Design - Purpose and Process", Intervoice, Inc. Microsoft.NET Speech Technologies Technical Articles. (2003)
4. Smashing magazine "12 Useful Techniques For Good User Interface Design", (2009), Retrieved 28 May 2010 from http://www.smashingmagazine.com/2009/01/19/12-useful-techniques-for-good-user-interface-design-in-web-applications/
5. Azeta A. A., Ayo C. K., Atayero A. A. and Ikhu-Omoregbe N. A., A Case-Based Reasoning Approach for Speech-Enabled e-Learning System", *2nd IEEE International Conference on Adaptive Science & Technology (ICAST).* December 14 – 16, 2009, Accra Ghana.
6. Gallivan P., Hong Q., Jordan L., Li E., Mathew G., Mulyani Y., Visokey P. & Tappert C., "VoiceXML Absentee System, *Proceedings of MASPLAS 2002*. The Mid-Atlantic Student Workshop on Programming Languages and Systems Pace University. Retrieved online 10th January 2010 from http://csis.pace.edu/csis/masplas/p10.pdf

7. Motiwalla, L.F. and Qin J." Enhancing Mobile Learning Using Speech Recognition Technologies: A Case Study". Eighth World Congress on the Management of eBusiness, 2007. (WCMeB 2007). Volume , Issue , 11-13 July 2007 Page(s):18 – 18

8. *Raghuraman M. B.,"Design and implementation of V-HELP system – a voice-enabled web application for the visually impaired. A Master's project presented to the faculty of the graduate college in the University of Nebraska in partial fulfillment of requirements for the degree of Master of Science. Major: Computer Science. August 2004. Retrieved 21 March 2010 from http://hcc.unl.edu/netgroup/alumnidocs/mukesh.MS.pdf*

9. Zambrano E. C., "Framework For Ubiquitous and Voice Enabled Web Applications Development: A Transport Information Systems Test – Bed, Doctoral Thesis Dissertation, Departamento de Informática, Universidad de Valencia, Spain, 2004.

10. Pakucs, B., "SesaME: A Framework for Personalised and Adaptive Speech Interfaces", In: Proceedings of EACL-03 Workshop on Dialogue Systems: Interaction, Adaptation and Styles of Management, Budapest, Hungary (2003) 95–102.

11. Yankelovich, N. "Designing Effective Speech Interfaces", John Wiley & Sons, Inc. An Interview edited by edited by Susan Weinschenk and Dean T. Barker (2000)

12. Chang S. E. (2007), "An Adaptive and Voice-enabled Pervasive web System", International Journal of Enterprise Information Systems (2007) ,Volume 3, Issue 4. pp. 69-83 IGI publishing