computing@tanet.edu.te.ua
www.tanet.edu.te.ua/computing

ISSN 1727-6209
International Scientific
Journal of Computing

VOICEXML-APPLICATIONS FOR E-COMMERCE AND E-LEARNING

Peter J. A. Reusch [1)], Bastian Stoll [2)], Daniel Studnik [3)], Jцrg Swade [3)]

[1)] University of Applied Sciences FH-Dortmund Germany, Peter.Reusch@FH-Dortmund.de
[2)] University of Applied Sciences FH-Dortmund Germany, Bastian.Stoll@FH-Dortmund.de
[3)] University of Applied Sciences FH-Dortmund Germany

Abstract: *VoiceXML is a language of the W3C to create voice-user interfaces, particularly for the telephone. It uses speech recognition and touchtone (DTMF keypad) for input, and pre-recorded audio and text-to-speech synthesis (TTS) for output. The text-to-speech synthesis feature of advanced VoiceXML tools like WebSphere opens new perspectives for e-commerce and e-learning. We are no longer restricted to pre-recorded audio but can bring any text to the ear of the user – a user that could be visually impaired and needs a voice channel to communicate – or a user who can read but who prefers to listen. VoiceXML-applications have been implemented by the authors to support e-commerce (selection of commodities from catalogues) and user guides for hardware (mobile phones, etc.) and software systems (MS project, etc.). New contributions to e-learning are offered.*

Keywords: *VoiceXML, XHTML+Voice, speech recognition and speech synthesis, e-commerce, e-learning.*

1. INTRODUCTION

Voice-enabled applications

Voice-enabled applications are available in several areas, for example:

Information providers

Voice response facilities are used for various kinds of information over the phone: time, weather, horoscopes, lottery results, sports events, news, exchange rates, and so on.

Financial institutions

A bank can let its customers access their account balances, obtain information on interest rates and mortgages, calculate loan payments, or transfer funds, all using voice response applications. An application can also call customers to inquire about transactions such as renewing a Certificate of Deposit.

Using a voice response application, brokerage firms can make current stock prices, quotations, and portfolio balances available over the telephone. Clients can perform complex transactions without the intervention of a broker. When a broker's advice is required, the application can transfer the call.

Educational institutions

A voice response application can provide information about class schedules, availability, and course content. Students can register using the telephone, and the application that handles the registration process can also update the database containing enrolment information. A voice response application can call students to inform them of schedule changes or openings in a class for which enrolment had been closed.

Voice-enabled applications will be improved and expanded significantly in the future, because there is a growing demand – and new technologies are available, that support the integration of voice-enabled functions in more and more applications.

In the past, voice-enabled applications were based on pre-recorded audio. New information models and speech-synthesis tools will boost the development of voice-enabled applications.

This kind of transformation allows new voice-enabled applications and supports visually impaired users significantly for example by offering user guides – an application described below.

2. VOICEXML

The **Voice** E**x**tensible **M**arkup **L**anguage VoiceXML is an XML-based language of the W3C to create voice-user interfaces, the latest version is available at: http://www.w3.org/TR/voicexml20/

VoiceXML is designed to create audio dialogs. Its major goal is to bring the advantages of Web-based development and content delivery to interactive voice response applications

VoiceXML describes the human-machine interaction provided by voice response systems, which includes:

- Output of synthesized speech (text-to-speech).
- Output of audio files.
- Recognition of spoken input.
- Recognition of DTMF[1] input.
- Recording of spoken input.
- Telephony features such as call transfer and disconnect.

The XML root element of a VoiceXML file is <vxml>, which is mainly a container for *dialogs*. There are two types of dialogs: *forms* and *menus*. Forms present information and gather input; menus offer choices of what to do next.

Figure 1 shows part of the definition of a VoiceXML menu according to the XML Schema Language XSD, available at http://www.w3.org/TR/voicexml20/vxml.xsd

```
<xsd:element name="menu">
  <xsd:complexType mixed="true">
    <xsd:choice minOccurs="0"
        maxOccurs="unbounded">
      <xsd:group ref="audio" />
        <xsd:element ref="choice" />
        <xsd:group ref="event.handler" />
        <xsd:element ref="property" />
        <xsd:element ref="prompt" />
    </xsd:choice>
        <xsd:attribute name="id" type="xsd:ID" />
          <xsd:attributeGroup
              ref="GrammarScope.attrib" />
          <xsd:attributeGroup
                  ref="Accept.attrib"/>
          <xsd:attribute name="dtmf"
            type="Boolean.datatype"
            default="false" />
..</xsd:complexType>
</xsd:element>
```

Fig. 1 – XML Schema Definition of a VoiceXML Menu

VoiceXML is a key used to transfer text to speech or database entries to speech in a very flexible way. Any text with its index of contents can be transferred from the document file into forms and menus of VoiceXML files that can be read out by text-to-speech synthesis tools like WebSphere.

The following figure shows the core architecture of VoiceXML applications.

A *document server* processes *requests* from a client application, *the VoiceXML Interpreter*, through the *VoiceXML interpreter context*. The server produces *VoiceXML documents* in reply, which are processed by the VoiceXML Interpreter. The VoiceXML interpreter context may monitor user inputs in parallel with the VoiceXML interpreter. The *implementation platform* is controlled by the VoiceXML interpreter context and by the VoiceXML interpreter.[2]

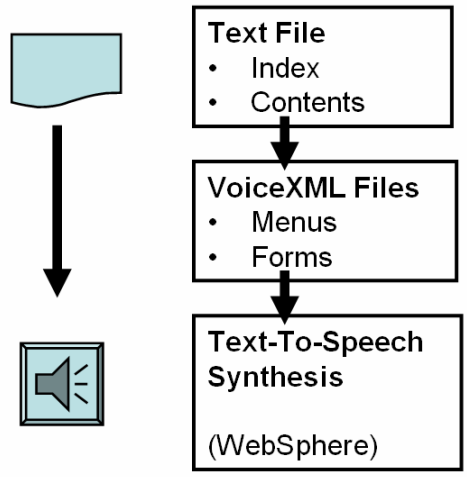


Fig. 2 – Text to Speech Transformation through VoiceXML

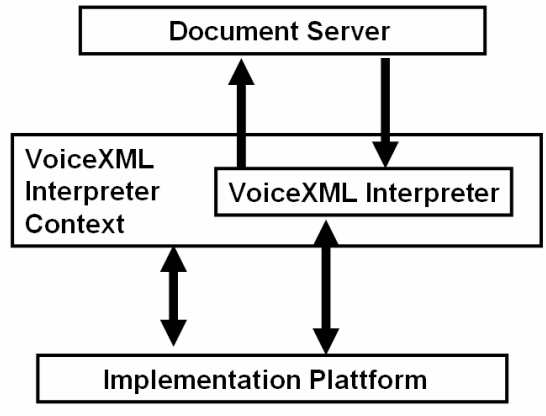


Fig. 3 – Architectural Model of VoiceXML

3. VOICEXML-BASED USER GUIDE FOR A MOBILE PHONE

All kinds of user guides can be transferred into voice-enabled applications. Here a voice-enabled user guide for the Siemens SL55 mobile phone will be described.

The following figure shows the mobile phone Siemens SL55. The user guide is available in a printed version attached to the product, a user guide is also available on web sites[3] and can be

[1] DTMF (Dual Tone Multi-Frequency)
Touch-tone or push-button dialing. Pushing a button on a telephone keypad generates a sound that is a combination of two tones, one high frequency and the other low frequency

[2] http://www.voicexml.org/specs/VoiceXML-100.pdf

[3] http://communications.siemens.com/cds/frontdoor/0,2241,hq_en_0_15799_rArNrNrNrN,00.html

downloaded[4]. It is no problem to study this user guide – 73 pages – if you can read. If you are visually impaired, you need others that can help – or a tool that can transform the document to speech.

Fig. 4 – Siemens SL55 Mobile Phone

The authors have implemented a voice-enabled user guide for this mobile phone. VoiceXML vxml-files include the information and menus that allow the user to navigate through the user guide.

The following figure shows the files of the user guide, starting with the vxml-file of the main menu – main_menu.vxml - and other vxml-files with specific information on selected items. For all vxml-files there are corresponding grammar files.

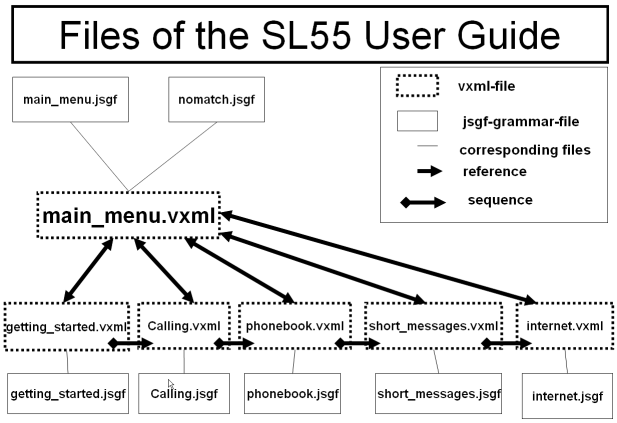


Fig. 5 – Files of the VoiceXML-based User Guide for the Siemens SL55 Mobile Phone

The following figure shows a part of main_menu.vxml (Fig. 6).

This file includes a text to welcome the user who is listening to the user guide and it includes the menu for the first choices:

- Getting started
- Calling
- Phonebook
- Short Messages
- Internet
- Exit

```
<form id="selection">
 <field name="main_menu">
 <grammar
src="main_menu.jsgf"></grammar>
<option dtmf="1" value="getting_started">
Getting started</option>
<option dtmf="2" value="Calling">
Calling</option>
<option dtmf="3" value="Phonebook">
Phonebook </option>
<option dtmf="4" value="Short_Messages">
Short Messages </option>
<option dtmf="5" value="Internet"> Internet
</option>
<option dtmf="6" value="Exit"> Exit
</option>
<prompt bargein="true">

Welcome to the voice-controlled user guide for
your mobile phone SL 55.
If you are familiar with the system you can get
direct access to the main menu. Please interrupt
me now and tell me, what kind of help you need.
Below you will find a menu with 6 choices. In
this menu the most important control elements
for your mobile phone are explained.
You are free to switch between the menu
items, whenever you like. If you would like to
listen to some menu items again, just tell me.
Now, take your time and listen to the
instructions carefully.

I suggest you listen to the getting started
instructions first.
There you get an overview of your mobile
phone.
```

Fig. 6– Part of main_menu.vxml

The menu is also included in the corresponding grammar file

```
#JSGF V1.0 iso-8859-1;
grammar main_menu;
public <main_menu> = Getting started |
Phonebook | Calling | Short Messages | Internet |
Exit ;
```

Fig. 7 Grammar file main_menu.jsgf

The VoiceXML user guide was implemented on IBMrs Websphere. The WebSphere Voice Server includes[5]:

[4] http://communications.siemens.com/repository/169/16990/sl55_userguide_3_comaucacnhkieisrllibmanzpaksgzatwthaeukusye_eng.pdf

[5] http://www-306.ibm.com/software/pervasive/voice_server/about/

- A speech-recognition (ASR) engine (speech-to-text software) that detects and recognizes words that are spoken over a telephone, then passes those words as text to an application.
- A text-to-speech (TTS) engine (voice-synthesis software) that synthesizes speech from application text for play-back over a telephone.
- Concatenate to TTS text.
- Connection to WebSphere Voice Response for AIX.
- Development tools, including support for applications that are written in VoiceXML, Java™ Beans, or C.

The following figure shows the structure of the WebSphere Voice Server.

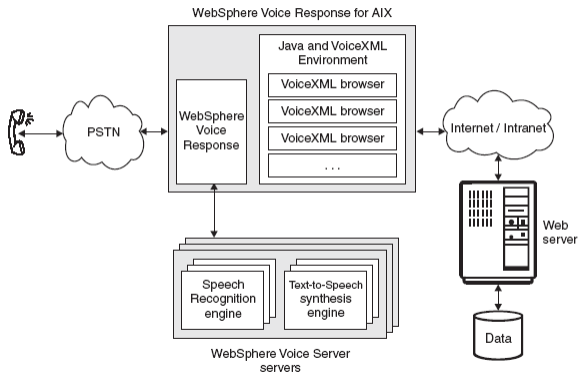


Fig. 8: WebSphere Voice Server[6]

And the next figure shows the screen of WebSphere while running the voice-enabled used guide for SL55. The user interaction is done by microphone and loudspeaker.

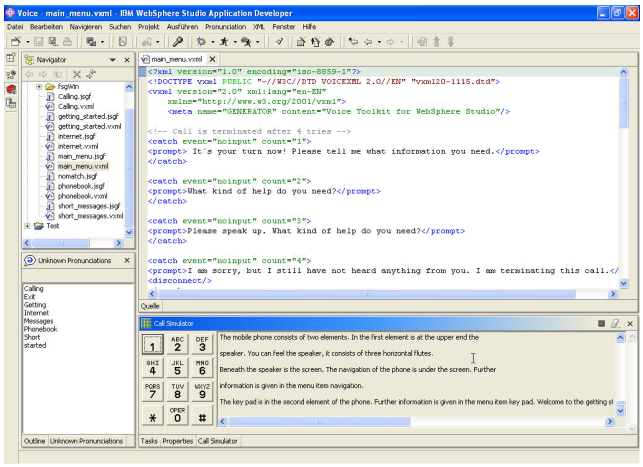


Fig. 9: WebSphere – SL55 User Guide Running

4. VOICEXML IN MULTIMODAL APPLICATIONS

VoiceXML is an XML-based language. Therefore it has common ground with the W3C-Standard XHTML, which is also based on XML. In cooperation IBM, Opera and Motorola tried to integrate both technologies and submitted a proposal for a new standard, XHTML+Voice, to the W3C. With XHTML+Voice multimodal web pages can be created. Multimodal means that information is provided in many different modes, such as text, speech or touch screen. In this case, it means that the user can decide to use the visual or the speech interface for input or output of information.

XHTML+Voice is a member of the XHTML family of document types, as specified by the XHTML Modularization. The XHTML Modularization extends XHTML with a modularized subset of VoiceXML 2.0, the XML Events module, and a module containing a small number of attribute extensions to both XHTML and VoiceXML. The modularized subset of VoiceXML contains nearly all important functions of VoiceXML, except for the only- telephony ones.

Here is a short example of how XHTML+Voice works:

```
<?xml version="1.0"?>
<html
xmlns="http://www.w3.org/1999/xhtml"
xmlns:vxml="http://www.w3.org/2001/vxml"
xmlns:ev="http://www.w3.org/2001/xml-events"
xmlns:xv="http://www.voicexml.org/2002/xhtml+voice"
>
  <head>
    <title>XHTML+Voice Example</title>
    <!-- voice handler -->
    <vxml:form id="sayHello">
      <vxml:block><vxml:prompt xv:src="#hello"/>
      </vxml:block>
    </vxml:form>
  </head>
  <body>
    <h1>XHTML+Voice Example</h1>
    <p id="hello" ev:event="click" ev:handler="#sayHello">
      Hello World!
    </p>
  </body>
</html>
```

Fig. 10: XHTML+Voice example[7]

In an XHTML+Voice environment, the top level VoiceXML element is the *<form>*-element. It is placed in the *<head>* of the XHTML file and contains the VoiceXML programme code. The *<form>*-element can be identified by an *id*, in this case *sayHello*.

In order to execute the VoiceXML code, the *<form>*element has to be activated by an event. In this example, it is a click event, which is triggered by the user.

Within the VoiceXML *<form>*-element, the text obtained from the same paragraph that activated the

[6] http://publibfp.boulder.ibm.com/epubs/pdf/c3463792.pdf

[7] http://www.voicexml.org/specs/multimodal/x+v/12/

form is synthesized. The speech output is *Hello World*!

This example shows the interaction of the event and the voice technology. Of course, this can be used to create very complex applications while considering the following scenario:

XHTML uses forms for the interaction with users. The following figure shows a sample.

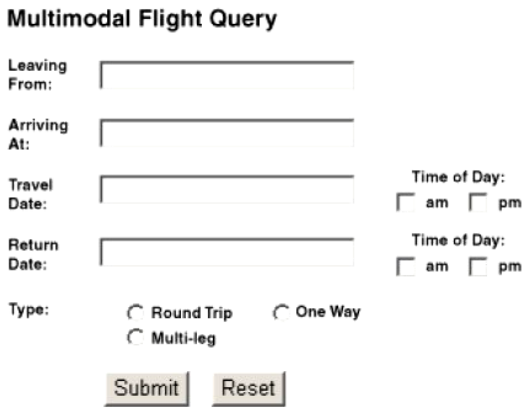


Fig. 11: XHTML form[8]

The form consists of textboxes, radio-buttons and checkboxes.

In this context XHTML+Voice enables the user to decide how to input his data. This can be done either by keyboard and mouse input or by using speech.

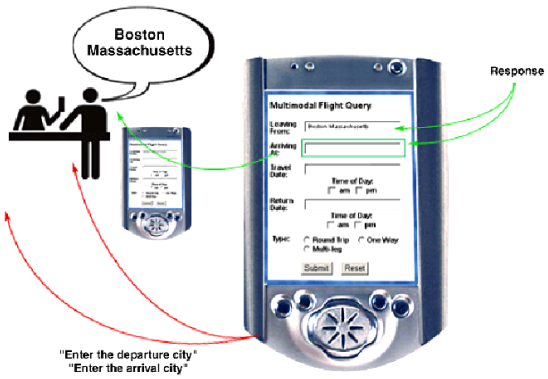


Fig. 12: Multimodal scenario on mobile devices

When thinking about the use of mobile devices, speech input can be a practical instrument for the user to fill the fields. Because sometimes it can be very complicated and time-consuming to type many words on a mobile's keypad.

[8] ftp://ftp.software.ibm.com/software/pervasive/info/multimodal/XHTML_voice_programmers_guide.pdf

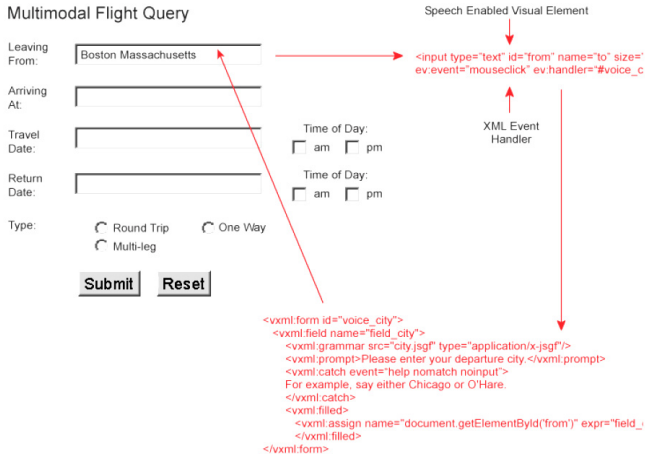


Fig. 13: Multimodal scenario

Considering the first XHTML+Voice example (Fig. 10), it is the event technology that activates the VoiceXML <*form*> element (Fig. 13).

In the same way, a voice output is generated by the system when clicking in a text field. The system output is called *prompt* and requests the user to make an input.

The speech input is understood by the system by use of the corresponding grammar element. It contains all suitable words that fit to the request. If there is any correspondence, the identified word or phrase is displayed in the text field, or the appropriate radio-button is activated.

This is possible, because the visual and the speech interface work closely together.

It is done by the element xv:sync.

```
<xv:sync xv:field="#voice_field1" xv:input="textfield"/>
<xv:sync xv:field="#voice_field2" xv:input="radio-button"/>
<xv:sync xv:field="#voice_confirm" xv:input="submitButton"/>
```

Fig. 14: Use of the xv:sync - element

This element exclusively exists within the XHTML+Voice – profile and synchronizes the visual and the speech interface. For example, if a user enters a word using speech while the text field is active, this word is also displayed graphically within the text field element.

As mentioned before, XHTML+Voice is based on the XHTML standard. For this reason it is also possible to integrate it into a PHP and MySQL environment. The advantage is that the powerful database functions can be used in combination with the voice functions not only to create multimodal websites, but also complex multimodal applications.

The authors created an application that can help students to train how to use MS Project. Synthetic speech output is generated that explains for example how to create a new project in MS Project.

The following figure shows such an example project.

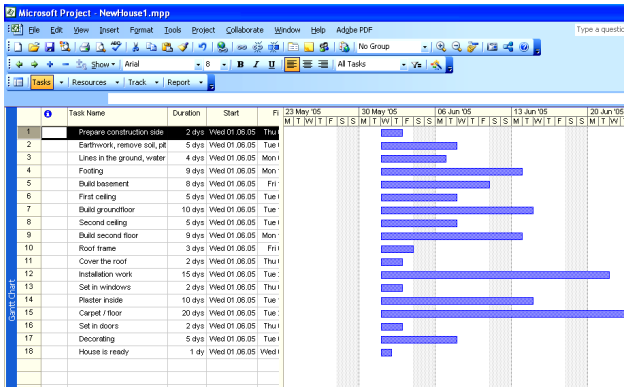


Fig. 15: Example project in MS Project

Students are able to work on this project while listening at the same time to the explanations from a synthesized voice.

To start the voice application the following screen is used.

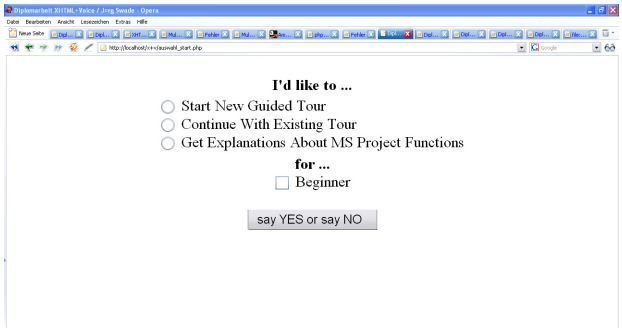


Fig. 16: Multimodal dialog

The user is prompted to make an input.

He is able to fill this form by using a keyboard and a mouse or by using speech. When all fields are filled, the system asks for a confirmation. This can be done by the user by saying *yes* or *no*.

The field *beginner* indicates that the application can process at a beginner and normal speed. The radio button *Continue With Existing Tour* shows that a user can continue with the application at the same point where he interrupted the tour. This is possible, because the current stage of the application is stored in the database.

If the radio-button *Get Explanations About MS Project Functions* is selected, the application changes to another screen, which is displayed in the next figure:

This text field can also be used for speech input. The user is able to change the current speed or exit the application. But the most powerful function of this dialog is to let the user name a function of MS Project for which he wants an explanation. An example could be: "Please tell me how to link tasks." This example shows that users are not only able to input words or phrases, but whole sentences.

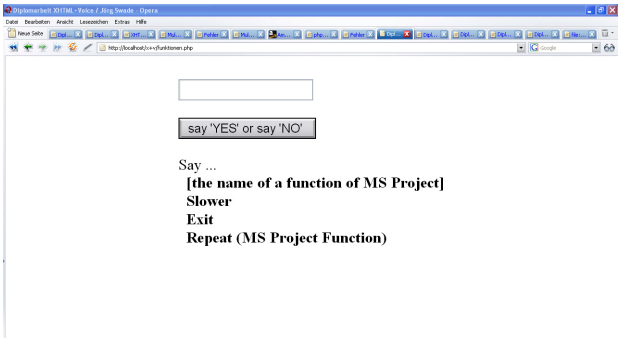


Fig. 17: Text field speech input

The user has the impression not only to navigate through a website but to be able to talk to the application.

5. VOICEXML FOR E-LEARNING

Voice-enabled applications can support e-learning in many ways. They can open e-learning systems to visually impaired users. As mentioned before – Fig. 2 – any document can be transferred through VoiceXML to speech-synthesis tools and the user can listen to it.

An equally important aspect is the opportunity to navigate through the "document" using VoiceXML menus that can be derived from the index of the original document automatically. This means that visually impaired users will be able to access a lot of e-learning applications in the future.

VoiceXML applications can be used through computers at home or in computer labs, but also through the telephone – without any computer on the part of the user. The e-learning application in this case is based upon a server with access through the telephone. In our days of telephone flat rates we can sit anywhere with our telephone and can listen to e-learning contributions.

On the other hand voice-enabled applications add new channels for everyone. We have our senses. We can see, hear, taste, touch and smell. Our ears can support e-learning. Today almost all speech components of e-learning systems are based upon pre-recorded audios. We use audibooks and "talking" dictionaries. Using VoiceXML we can bring much more to the ear for everyone.

6. CONCLUSION

VoiceXML applications can be established in an efficient way – also in new areas like user guides or e-learning. The applications are very flexible and can be derived from existing documents or data bases. The VoiceXML applications are available through computers or through the telephone, which is a universal means of communication.

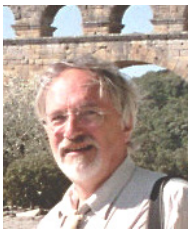
7. REFERENCES

[1]. Reusch, Peter J. A.; Reusch, Pascal: Classifications of Products and Services to Support Business Process Engineerung and e-Commerce, *International Scientific Journal of Computing,* Vol. 2, Issue 2, 133-140, 2003.

[2]. Stoll, Bastian: *Anwendung von VoiceXML bei sprachbasierter Arbeit mit Katalogen erstellt mit IBM Websphere Studio*, Diploma Thesis, Dortmund 2004.

[3]. Studnik, Daniel: *Erstellung einer sprachgesteuerten Bedienungsanleitung fьr ein Mobiltelefon mit der Programmiersprache Voice XML 2.0,* Project Thesis, Dortmund 2004.

[4]. Swade, Jцrg: *Entwicklung einer multimodalen, sprachbasierten Web-Based-Training – Anwendung fьr MS Project*, Diploma Thesis, Dortmund 2006.



Prof. Peter Reusch *was born in Germany in 1950. In 1976 he received the PhD degree in Computer Science from the University of Bonn. He is honorary doctor of the State Economic University in Minsk and of the University of Latvia in Riga. At present he works as Professor at the University of Applied Sciences in Dortmund and is course director of the European Masters in Project Management – EuroMPM. He is chair of the international consortium implementing the EuroMPM in Trondheim, Paris/Lille, Zaragoza/Rioja, Ljubljana/Maribor, GieЯen-Friedberg and Dortmund. He is chair of the Scientific Advisory Board of the e-Commerce Group at the Cologne Institute for Business Research. He is a member of the IDAACS Group. He teaches and makes contributions on research and development in several institutes in various countries.*

Bastian Stoll *graduated in 2004 from the University of Applied Sciences in Dortmund and works as assistant at that university within the EuroMPM-Programme.*

Danial Studnik *graduated in 2005 from the University of Applied Sciences in Dortmund.*

Jцrg Swade *graduated in 2006 from the University of Applied Sciences in Dortmund and started his own company based upon multimodal applications.*