# VoiceXML example

## Creating the welcome message

In this VoiceXML example, you will be creating an application that gives you a selection to choose from. Once you make a selection, you are taken to the appropriate document or dialog. In this section, you will create the main greeting message of the application. In the code below, the user hears a "welcome" message and is then given a list of choices from the main menu. The <goto> element is used to skip to the menu section.

```
<?xml version="1.0"?>
<vxml version = "2.0" xmlns="http://www.w3.org/2001/vxml">
 <!-- user hears welcome the first time -->
 <form id="intro">
  <block>
   <audio>Welcome</audio>
   <goto next="#make_choice"/>
  </block>
 </form>
</vxml>
```

## Creating the menu

In this section you will create the menu for the VoiceXML application. The menu gives you four selections to choose from. DTMF is enabled, which allows for key presses by setting the dtmf attribute to 'true' in the <menu> element. The menu is created using the <choice> element and the next attribute specifies what document or anchor to go to. An event is caught, using the <catch> element, in case the user doesn't respond, says "help", or the input by the user is not recognized.

```
<?xml version="1.0"?>
<vxml version = "2.0" xmlns="http://www.w3.org/2001/vxml">
 <menu id="make_choice" dtmf="true">
  <prompt>
   <audio>Say survey, weather, news, or quit.</audio>
  </prompt>
  <!-- next attribute takes you to the appropriate documents or anchor
within the current -->
  <choice next="http://www.hostname.com/survey.vxml">survey</choice>
  <choice next="http://www.hostname.com/weather.vxml">weather</choice>
  <choice next="#news_report">news</choice>
  <choice next="#quit_app">quit</choice>
  <catch event="nomatch noinput help">
   <reprompt />
  </catch>
 </menu>
</vxml>
```

## Creating the anchors

In this section you will create the anchor sections that appear in the same document as the menu

section. The two choices, news and quit, go to the following sections:

```
<?xml version="1.0"?>
<vxml version = "2.0" xmlns="http://www.w3.org/2001/vxml">

 <!-- give the news -->
 <form id="news_report">
  <block>
   <audio>Sports news!
    Michael Jordan returned to the NBA hardcourt for the first time in
nearly 40 months.</audio>
   <goto next="#make_choice"/>
  </block>
 </form>

 <!-- quit the application -->
 <form id="quit_app">
  <block>
   <audio>Goodbye!</audio>
  </block>
 </form>
</vxml>
```

**Creating the weather document**

In this section you will create the document that gives the user weather information. Once the user says or selects weather from the menu, the following code is executed:

```
<?xml version="1.0"?>
<vxml version = "2.0" xmlns="http://www.w3.org/2001/vxml">
 <form id="weather_report">
  <block>
   <audio>
    It will be partly cloudy today.</audio>
   <goto next="http://www.hostname.com/main.vxml"/>
  </block>
 </form>
</vxml>
```

Creating the survey document The survey document prompts the user to respond to a question. Once the user completes the survey, the user is taken back to the main menu. In the code below the user can respond to the survey question by selecting one of the choices from the active grammar. Below is the active grammar list for the survey document:

```
<grammar>
<![CDATA[
[
[one excellent dtmf-1] {<option "excellent">}
[two good dtmf-2] {<option "good">}
[three fair dtmf-3] {<option "fair">}
[four poor dtmf-4] {<option "poor">}
```

```
[five help dtmf-5] {<option "help">}
]
]]>
</grammar>
```

The user is prompted with a question – if the user does not say anything, or if there is no match with the active grammar, the user is asked the survey question again. If the user asks for help, the user hears the help information and is asked the question again.

After successfully responding to the survey question, the code within the <filled> element is executed and the user hears a confirmation message and is returned to the main menu.

```
<!-- prompt the user what to do -->
<prompt>
 Welcome to the survey! What did you think of your
 instructor? If you don't know what to do, say help
 or push 5.
</prompt>
<!-- if no match with active grammar list then user prompted again. -->
<nomatch>
 What did you say?
 <reprompt/>
</nomatch>
<!-- executed if no input is provided by the user -->
<noinput>
 Please input something!
 <reprompt/>
</noinput>

<!-- executed if help is requested -->
<help>
 Please say what you think of your instructor.
 You can say excellent, good, fair, or poor.
 You can also push one for excellent, two for
 good, three for fair, and four for poor.
 <reprompt/>
</help>

<!-- on a successful input skip to the form confirmResponse -->
<filled>
 <goto next = "#confirmResponse"/>
</filled>
```

**The entire VoiceXML application**

- main.vxml
- survey.vxml
- weather.vxml

From:
https://wiki.voximal.com/ - **Voximal documentation**

Permanent link:
**https://wiki.voximal.com/doku.php?id=legacy:vxi_tutorial:example**

Last update: **2017/07/28 23:53**