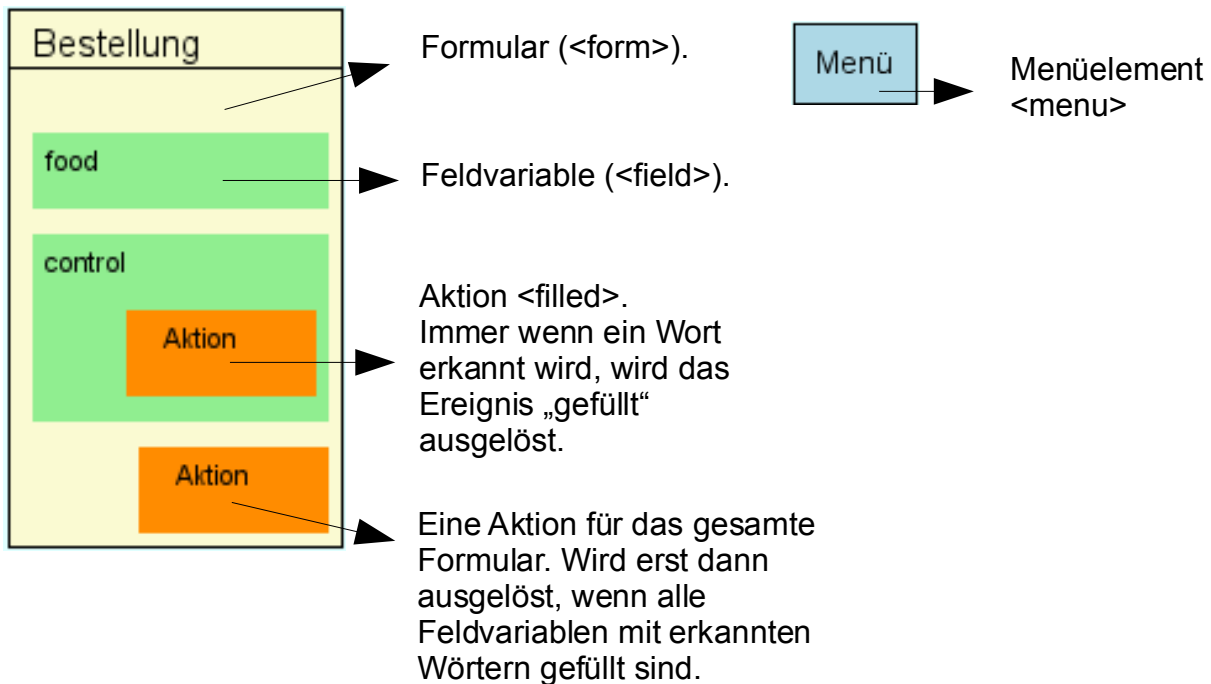




Graphische Elemente und deren Bedeutung für VXML



Allgemeine Erläuterungen zum Ausführen eines Sprachdialoges

Ein VXML-Dialog wird nach dem sogenannten Form-Interpretation-Algorithm (FIA) ausgeführt.

Der Algorithmus beginnt mit dem ersten VXML-Element. Handelt es sich dabei um ein Formular, wird dieses Formular zunächst analysiert. Dabei werden alle darin enthaltenen Feldvariablen initialisiert (auf „undefined“ gesetzt). Ein Formular gilt als „abgearbeitet“, wenn jede Feldvariable mit erkannten Wörtern/Zahlen gefüllt ist.

Innerhalb eines Formulars kann mit gewöhnlichen Kontrollstrukturen (if-else-elseif) gearbeitet werden, wenn beispielsweise ein Sprung in ein anderes Formular/Menü erwünscht wird.

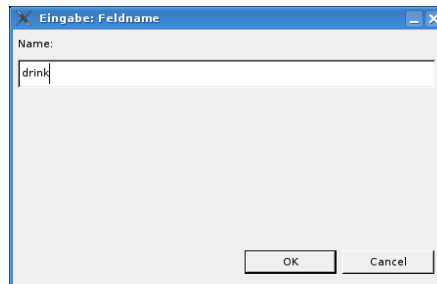
Vorgehensweise für das Hinzufügen einer Feldvariablen

Für das Hinzufügen einer weiteren Feldvariable muss zunächst ein Formular ausgewählt werden (Mausklick auf das entsprechende Formular).

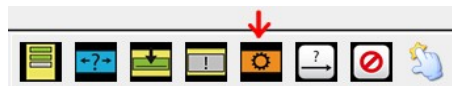
1. Drücke jetzt den Button „Feld“.



2. Es erscheint ein Fenster „Feldnamen eingeben“. Hier gibst Du nun den Namen der Feldvariablen ein und bestätigst mit „OK“.



3. Möchtest Du darüberhinaus noch eine Aktion hinzufügen, dann wähle Deinen zuletzt erstellten Feldnamen aus (Mausklick auf Feldname „drink“).
4. Drücke jetzt den Button „Aktion“.



5. Fertig.

Erstellen von Formularübergängen

Oft werden Sprünge (goto's) benötigt. Zum Beispiel wenn aus einem Menü ein bestimmtes Formular angesteuert werden muss. Werden solche Sprünge erforderlich, muss wie folgt vorgegangen werden.

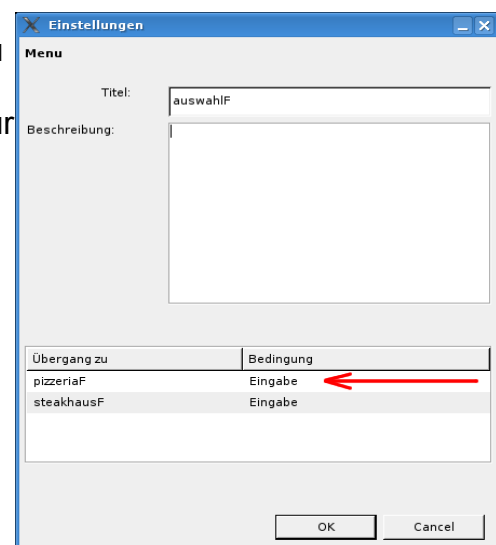
1. Wähle das Element aus, von dem Du springen möchtest. Danach drücken einmal den Übergangsbutter.



2. Jetzt wähle das Element aus (einmal anklicken), zu dem Du springen möchtest. Es wird ein Pfeil mit der Beschriftung „Eingabe“ angezeigt.
3. Klicke nun doppelt auf das Element von dem Du gesprungen bist. Es wird nun im Formular (oder Menüelement), das Du bearbeitest angezeigt, für welche erkannten Wörter, welches Formular angesprungen wird.

Wenn Du nun doppelt auf „Eingabe“ klickst, dann kannst Du stattdessen das zu erkennende Wort eingeben.

Alternativ kannst Du auch direkt auf das im Pfeil befindliche Rechteck klicken und das zu erkennende Wort direkt eingeben.





Umwandeln eines Modells und Implementieren einer eigenen Anwendung

Wenn Dein Modell fertig ist, dann beginnst Du Deine Implementation durch drücken des Buttons „Code Vorlage“.



Achtung! Der gesamte VXML-Code auf der rechten Seite wird synchronisiert. Ein entsprechender Hinweis wird Dich darauf aufmerksam machen.

Nach dem Umwandeln siehst Du, dass inES die Feldvariable Deinem Quelltext hinzugefügt hat.

Ein Beispiel:

Das Hinzufügen einer Feldvariable „drink“ zur Bestellung:

1. Gehe zunächst wie im Abschnitt „Vorgehensweise für das Hinzufügen einer Feldvariablen“ vor.

```
<vxml>
<form id="Bestellung">
  <field name="food">
    <prompt>Hallo. Hier können Sie etwas zu essen und trinken bestellen . Wä
    </prompt>
    <option value=" pizza ">Pizza</option>
    <option value=" hot dog ">Hot Dog</option>
    <option value=" hamburger ">Hamburger</option>
  </field>
  <field name="control">
    <prompt>Stimmt das?<enumerate/>
    </prompt>
    <option value=" yes ">ja</option>
    <option value=" no ">nein</option>
  </field>
  <filled>
    <if cond=" control == 'no'">
      <prompt>Sie sagten <value expr="control"/>
    </prompt>
    <clear namelist=" food control !/>
    <goto next="#Bestellung"/>
    </if>
  </filled>
</field>
</form>
<filled>
  <prompt>Danke, dass Sie <value expr="food"/> bestellt haben. Auf Wieder
</prompt>
</filled>
</vxml>
```

2. Jetzt drücke den Button „Code Vorlage“ in inES.



The screenshot shows the inES development environment. The left pane contains a code editor with the following XML code:

```
<field name="food">
  <prompt>Hallo. Hier können Sie etwas zu essen und trinken bestellen . Wä
  </prompt>
  <option value=" pizza ">Pizza</option>
  <option value=" hot dog ">Hot Dog</option>
  <option value=" hamburger ">Hamburger</option>
</field>
<field name="control">
  <prompt>Stimmt das?<enumerate/>
  </prompt>
  <option value=" yes ">ja</option>
  <option value=" no ">nein</option>
</field>
  <!-- if control == 'no' -->
  <prompt>Sie sagten <value expr="control"/>
  </prompt>
  <clear namelist=" food control "/>
  <goto next="#Bestellung"/>
</if>
</field>
<field name="drink">
  <!--[TODO] Implementiere hier Deine Sprachausgaben und Variablen.-->
  </field>
  <!--[TODO] Implementiere hier Deine Ereignisse.-->
</field>
</field>
  <prompt>Danke, dass Sie <value expr=" food"/> bestellt haben. Auf Wiede
  </prompt>
</field>
</form>
</vxml>
```

The right pane shows a visual model titled "Bestellung". It contains three elements: "food", "control", and "drink". Each element has an "Aktion" button next to it. The "food" element is highlighted in green, "control" in light green, and "drink" in light orange.

3. Du siehst jetzt die neu hinzugefügte Variable „drink“. Die Kommentare weisen Dich darauf hin, welche Aufgabe Dir als Entwickler bleibt.

Alternativ kannst Du selbstverständlich auch ohne die Aktualisierung des Modells Änderungen am VXML-Code vornehmen.

Anmerkung

Du musst nicht andauernd in reinem Hochdeutsch sprechen um den Sprachdialog zu testen. Für schnelle Tests empfehle ich die Spracherkennung zu deaktivieren. Das kann sowohl während der Ausführungsphase als auch über Einstellungen->Spracherkennung erledigt werden.

Viel Spaß beim Ausprobieren!



Übersicht über die von inES unterstützten VXML-Befehle

| | | | | | |
|----------|--|---|----------|--|--|
| <vxml> | | Erstellung eines VXML-Dialoges | <form> | | Erstellung eines Formulars |
| <menu> | | Erstellung eines VXML-Menüs | <field> | | Feldvariable deklarieren. Attribute: name: Der Name der Feldvariablen. Beispiel: <field name="drink"> |
| <choice> | | Anwendbar innerhalb eines Menüs. Dient zur Erstellung einer Auswahl. Beispiel: <choice next="#animalsF"> animals</choice> <choice next="#citiesF"> cities</choice> Wenn ein Wort wie cities erkannt wurde, dann wird zum Formular citiesF gesprungen. | <option> | | Auswählen von Optionen. Wird oft in Verbindung mit prompt benutzt. Beispiel: <option>yes</option> |
| <prompt> | | Ausgeben von Text. Kann sowohl bei Menüs als auch bei Formularen benutzt werden. | <block> | | Anwendbar innerhalb eines Formulars. Wird benutzt wenn mehrere Anweisungen hintereinander ausgeführt werden sollen, ohne das ein Ereignis vorliegt. |
| <clear> | | Löschen von Variableninhalten, bzw. setzt die Variablen wieder auf „undefined“. Beispiel: <clear namelist="control"/> Setzt control wieder auf „undefined“. | <value> | | Wird benutzt um den Inhalt einer Variable anzuzeigen. Beispiel: <value expr="food"/> Gibt den Inhalt der Variable food aus. |
| <if> | | Wird in Formularen benutzt. Benutzung wie herkömmliche if-Anweisungen. Kann mit <elseif> und <else> kombiniert werden. Beispiel: <if cond="food=='Hamburger'"> <!-- TU WAS!!! --> </if> Stringvergleiche müssen in ' gesetzt werden. | <var> | | Definieren von Variablen. Beispiel: <var name="kosten" /> Definiert die Variable kosten. |
| <assign> | | Weist Variablen einen Wert zu. Beispiel: <assign name="kosten" expr="30"/> Weist der Variable kosten die Zahl 30 zu. | <goto> | | Springt zu einem Formular / Menü, oder zu einem Feld innerhalb eines Formulars. Beispiel: <goto next="#bestellungF" /> Spring zum Formular bestellungF <goto nextitem="field" /> Spring zum Feld field innerhalb eines Formulars. |

Logische (Boolesche) Operatoren in VXML

| Operator | Bedeutung | Beispiel |
|------------|--|--|
| | Logisches „oder“ | <if cond="zahl1 == '1' zahl2 == '2'"> <!-- Tu was --> </if> |
| && | Logisches „und“ & steht für „ampersand“ was dem &-Zeichen entspricht. | <if cond="zahl1 == '1' && zahl2 == '2'"> <!-- Tu was --> </if> |
| XOR | Achtung! VXML unterstützt, weil Javascript benutzt wird, kein XOR. | Damit trotzdem ein XOR möglich wird, muss die Bedingung ähnlich dieser formuliert werden: <if cond="(zahl1 == '1' && zahl2 != '2') (zahl1 != '1' && zahl2 == '2') "> ... </if> Bedeutung: Entweder ist zahl1 „1“ oder „2“. |