Brandenburgische Technische Universität
Cottbus - Senftenberg

Lehrstuhl Kommunikationstechnik
Fakultät 3

Bachelorarbeit

über das Thema

Wizard-of-Oz-Methode Sprachsteuerung einer Musikbibliothek

wizard-of-oz-method / voice command for a music libary

Autor: Mike Stöhr
Matrikel-Nr.: 2904760
E-Mail: stoehmik@tu-cottbus.de

Betreuer: Prof.Dr.-Ing. habil. Matthias Wolff
Dr.-Ing. Ronald Römer

Abgabedatum: 15.09.2014

Selbstständigkeitserklärung

Der Verfasser erklärt, dass er die vorliegende Arbeit selbständig, ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt hat. Die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind ausnahmslos als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

....

Ort, Datum (Unterschrift)

I Kurzfassung

In dieser Arbeit wird es darum gehen, am Beispiel einer Sprachsteuerung für eine Musikbibliothek, die Wizard-of-Oz (WoO)-Methode genauer zu untersuchen, um später mit diesem Wissen ein benutzerfreundliches User Interface (UI) für Sprachsteuerungen entwickeln zu können. Dabei wird das Hauptaugenmerk darauf liegen, wie man das dafür notwendige Experiment vorbereitet, wie man die Aufgaben den Probanden stellen kann, wie solche Versuche ablaufen und was es zu beachten gibt, auch nach der Durchführung solch eines WoO-Experimentes. Bei dieser simulierten Mensch-Maschine-Kommunikation (MMK) soll erreicht werden, eine Schablone für dass Experiment (Aufbau, Aufgabenblätter etc.) zu entwickeln, aus denen weitere WoO-Experimente leichter und schneller abgeleitet werden können, damit man den Zeitraum zur Erstellung ähnlicher Experiment zukünftig verkürzen kann. Am Ende muss beurteilt werden, ob die erstellte Schablone von Aufgabenblättern und die Durchführung der einzelnen Schritte der WoO-Methode funktioniert und so für weitere Versuche und Entwicklungen übernommen werden kann. Zu dem wird in dieser Arbeit der räumliche und technische Aufbau des Experimentes detailliert dokumentiert sein, um diesen identisch im Sprachlabor des Lehrstuhls für Kommunikationstechnik wieder einrichten zu können.

Abstract

The point of this paper is to analyse the Wizard of Oz method, using, as an example, the voice command for a music library, in order to use the knowledge to be able to develop a user-friendly interface for voice commands. The main focus will be, how to prepare the necessary experiment, how to allocate responsibilities within the subject group, how such tests are conducted and what has to be considered, even after performing such an experiment. In this simulated human-machine communication the attempt will be made to develop a template for this experiment (structure, exercise sheets etc.), through which further Wizard of Oz experiments can be be carried out easier and faster, so that the time-frame for drawing up similar experiments can be shortened. In the final stages, it has to be assessed whether the constructed template of exercise sheets and the step by step execution of the Wizard of Oz method works and can be carried over for further tests and developments. Furthermore, this work will document in detail the physical and technical setup of the experiment, so that it can be reconstructed one to one in the language laboratory of the professorship for communication technology.

II Inhaltsverzeichnis

I Kurzfassung **I**

II Inhaltsverzeichnis **II**

III Abbildungsverzeichnis **IV**

IV Abkürzungsverzeichnis **V**

1 Einführung **1**
1.1 Motivation . . . 1
1.2 Aufbau der Arbeit . . . 1

2 Wizard-of-Oz-Methode **2**
2.1 Ziel von Wizard-of-OZ Versuchen . . . 2
2.2 Allgemeiner Aufbau von Wizard-of-Oz Versuchen . . . 3
2.3 Allgemeiner Ablauf von Wizard-of-Oz Versuchen . . . 3

3 Bewertung der Validität **5**
3.1 Kriterien nach Fraser und Gilbert . . . 5
3.2 Kriterien nach Green und andere (u.a.) . . . 5
3.3 Kriterien nach Steinfeld u.a. . . . 6

4 Media Player **8**

5 Experiment **10**
5.1 Vorbereitung . . . 10
5.2 Entwicklung der Aufgabenstellung . . . 11
5.2.1 Semiotisches Dreieck . . . 11
5.2.2 Erste Entwicklungsschritte . . . 12
5.2.3 Aufgabenblattversion 1 . . . 14
5.2.4 Aufgabenblattversion 2 . . . 16
5.3 Aufbau des Experimentes . . . 17
5.4 Systemausgabe . . . 20
5.5 Musik Datenbank . . . 21
5.6 Zielgruppe . . . 22
5.7 Durchführung . . . 22

6 Bewertung und Auswertung **24**
6.1 Stärken und Schwächen . . . 24
6.2 Auswertung vom Fragebogen . . . 25
6.3 Datenauswertung . . . 28

7 Fazit und Ausblick **31**

8 Quellenverzeichnis **33**

Anhang **I**

A Fragebogen Experiment **I**

B Handout Experiment **II**

C Erstversuche Aufgabenstellung (AGSt) **V**

D Fertige AGSt **VII**

E Weitere sprachgesteuerte Aufgaben fürs System **XI**

F Liste mit Schnelltastenauswahl **XII**

G Schaltplan Sprachlabor **XIII**

H Grammatik **XIV**

III Abbildungsverzeichnis

Abb. 1 Kombinationen Steinfeld 6
Abb. 2 Bibliotheksansicht 8
Abb. 3 Abspielansicht 9
Abb. 4 Windows-Media-Player (WMP) Designdarstellungen 9
Abb. 5 ERM adressierbare Attribute 11
Abb. 6 Darstellung semiotisches Dreieck 12
Abb. 7 Erstversuch Aufgabenstellung (1) 13
Abb. 8 Modell Syntagma-Paradigma 14
Abb. 9 Aufgabenstellung Variante 1 15
Abb. 10 Aufgabenstellung Variante 2 17
Abb. 11 Raumansicht Sprachlabor 18
Abb. 12 technischer Grundriss Sprachlabor 19
Abb. 13 Anschlüsse Soundkarte / Kopfhörerverteilverstärker 19
Abb. 14 Textausgabe Fehlermeldung 21
Abb. 15 XML-Anfrage Datenbank 21
Abb. 16 Problematisches Aufgabenblatt 10 25
Abb. 17 Problematisches Aufgabenblatt 7 26
Abb. 18 Gleiche AGSt in beiden Varianten 27
Abb. 19 Auszug Grammatik 29
Abb. 20 Aufgabenblatt 9 30
Abb. 21 Erstversuch 1 V
Abb. 22 Erstversuch 2 V
Abb. 23 Erstversuch 3 V
Abb. 24 Erstversuch 4 VI
Abb. 25 Erstversuch 5 VI
Abb. 26 Erstversuch 6 VI
Abb. 27 Aufgabenblatt 1 VII
Abb. 28 Aufgabenblatt 2 VII
Abb. 29 Aufgabenblatt 3 VII
Abb. 30 Aufgabenblatt 4 VIII
Abb. 31 Aufgabenblatt 5 VIII
Abb. 32 Aufgabenblatt 6 VIII
Abb. 33 Aufgabenblatt 7 IX
Abb. 34 Aufgabenblatt 8 IX
Abb. 35 Aufgabenblatt 9 IX
Abb. 36 Aufgabenblatt 10 X
Abb. 37 Aufgabenblatt 11 X
Abb. 38 Aufgabenblatt 12 X
Abb. 39 Aufgabenblatt 13 XI
Abb. 40 Aufgabenblatt 14 XI
Abb. 41 Grammatik XIV

IV Abkürzungsverzeichnis

UI User Interface
WoO Wizard-of-Oz
MMK Mensch-Maschine-Kommunikation
u.a. und andere
WMP Windows-Media-Player
ERM Entity-Relationship-Modell
KT Kommunikationstechnik
AGSt Aufgabenstellung
UVNC Ultra Virtual Network Computing
LCD liquid-crystal Display

1 Einführung

1.1 Motivation

Das sprach-basierte UI verzeichnete in den letzten Jahren einen zunehmenden Wachstum in der MMK und wird inzwischen eingesetzt, um mit unterschiedlichen technischen Geräten zu interagieren. Das hängt zu einem damit zusammen, dass mit der schnell wachsenden Technik und immer neuen Features in technischen Geräten, ein ausschließlich grafisches UI besonders für die älteren Generationen schwer zu bedienen ist, wie auch für andere Nutzer, die beispielsweise durch eine Sehbehinderung in der Interaktion zu sehr eingeschränkt werden. Zum anderen, dass diese Technik besonders in Bereichen eingesetzt wird, wo gestische oder manuelle Eingaben einfach nicht möglich sind, wie zum Beispiel in der Autoindustrie beim Bordcomputer. Auch in anderen technischen Umgebung sind sprach-basierte UI einfach angemessener und werden grafische Systeme, die oft an ihre Grenzen der Bedienbarkeit stoßen, zukünftig weiter ersetzen [PRC94]. Der einfache Vorteil von sprach-basierten UI ist, dass die Bedienung per Sprache natürlich von statten geht und kein großer Lernprozess von Nöten ist und somit, bis auf einzelne Ausnahmen, für jeden schnell und leicht umsetzbar ist. Doch oft ist es so, dass solche Sprachsysteme noch sehr fehlerbehaftet sind in ihrer Erkennung der menschlichen Sprache und die wohl viel wichtigere Komponente, dass diese Systeme ausschließlich versuchen die Eingaben zu *erkennen*, aber diese nicht wirklich *verstehen* können. Das Problem an der Erkennung hängt oft damit zusammen, dass in der Entwicklung, von einer sehr theoretischen Seite herangegangen wird und man die Eingabemöglichkeit zu sehr begrenzt und somit das Gerät nicht die vollständige Variation der Sprache *erkennen* kann, denn oft wird aus Zeit und Kostengründen auf den Einsatz der WoO-Methode verzichtet. In dieser Arbeit soll es also wirklich darum gehen, die ersten Entwicklungsschritte mit Hilfe der WoO-Methode für ein sprach-gesteuertes UI zu betrachten, um sie für folgende Experimente am Lehrstuhl bei ähnliche Ent-/Weiterentwicklungen nutzen zu können für die Entwicklung von sprach-basierten UI.

1.2 Aufbau der Arbeit

Am Anfang beschäftigt sich diese Arbeit mit dem allgemeinen Inhalten und den Nutzen der WoO-Methode. Im dritten Kapitel werden Kriterien von Wissenschaftlern betrachtet, die bei der Durchführung von eigenen WoO Experimenten aufgestellt wurden, um die Validität von solchen Versuchen zu bewerten. Weil das reale System noch nicht existiert und der WMP das geplante System ersetzt, betrachten wir kurz den Aufbau und die Funktionen einer der geläufigsten Player. In Kapitel 5 kommen wir zum wichtigsten Teil, dem Experiment, mit seiner Vorbereitung, dem Aufbau, die Durchführung und seiner Nachbearbeitung. Darauf folgt im sechsten Kapitel die Bewertung des entwickelten Szenarios, mit der Umsetzung der Aufgabenstellung und dem Ablauf der Durchführung. Außerdem erfolgt die Auswertung der gesammelten Daten und die Auswertung des Gespräches nach dem Versuch. Im siebten und damit auch letzten Kapitel erfolgt meine persönliche Bewertung des Szenarios und der Ausblick, wie sich die WoO Szenarien entwickeln.

2 Wizard-of-Oz-Methode

Unter der WoO-Methode, benannt durch John F. Kelley (1984) [Kel84] nach der Kindergeschichte "The Wonderful Wizard of Oz", versteht man eine Möglichkeit zum Designen eines UI. Ein noch nicht fertig existierendes UI kann durch den Menschen simuliert werden und somit können geplante Funktionen vor der Implementierung auf ihre Bedienbarkeit getestet werden. Damit kann in einem sehr frühem Entwicklungsstadium schon festgestellt werden, wie gut bedienbar das geplante System ist und gegebenenfalls schon früh optimiert werden.

2.1 Ziel von Wizard-of-OZ Versuchen

Vor der programmiertechnischen Umsetzung einer Benutzeroberfläche, muss ein Entwurf gemacht werden, wie das UI letztendlich aussehen soll. Es gibt diverse Regeln zum erfolgreichen erstellen eines grafischen User Interfaces und auch Grundregeln für die allgemeine Entwicklung von UI, die beachtet werden sollten (unnötige Aktionen für den Benutzer vermeiden, ein einheitliches Bedienkonzept, Konsistenz im Programmaufbau usw. [Rud06][Ste]). Dennoch geben solche Regeln keine Garantie für eine gut bedienbare Oberfläche, weil das Verhalten von Menschen nicht vorhersehbar ist. Ein gutes Beispiel dafür ist, dass viele Menschen im Umgang mit einer sprach-gesteuerten MMK oft dazu neigen kurze Befehle oder sogar nur einzelne Worte zu verwenden, um ihre Anforderungen ans System zu stellen, obwohl viele Programme schon im Stande sind ganze gesprochene Sätze zu verstehen. Auch in dem hier betrachteten Beispiel soll es um solch eine Umsetzung, die komplette Sätze versteht, gehen und es wird später auch betrachtet, wie die Probanden mit dem System umgegangen sind. Ein anderes Beispiel ist die Funktionsfülle von technischen Geräten, wie zum Beispiel in Mobiltelefonen. Da werden Telefone mit einer Vielzahl von Funktionen umworben und auch wenn der Benutzer nicht alle davon brauch oder gar nicht genau weiß wie diese zu bedienen sind, werden oft die Geräte gewählt die mehr Funktionen bieten, als Geräte die leichter zu bedienen wären mit weniger Funktionen. An solchen Beispielen sieht man, das kein UI entwickelt werden sollte mit einer rein theoretische Herangehensweise. Genau an diesem Punkt kommt die WoO-Methode zum Einsatz, um sich darüber Klarheit zu verschaffen, inwiefern das eigene Benutzersystem beherrschbar ist bzw. wie der Benutzer am liebsten mit dem Programm interagieren möchte. Damit lässt sich vor einer unnötigen Programmierarbeit das System schon so anpassen und testen, damit am Ende nicht das komplette Programm umgeschrieben werden muss und man immer wieder Rückschritte macht.

Das Hauptziel solcher Versuche muss es also sein, alle geplanten Funktionen zu testen auf ihre Bedienbarkeit und um ein genaues Bild davon zu bekommen, wie der Mensch mit dem System umgeht und welche Anforderungen er an selbiges stellt.

Ein weiteres Ziel ist es, Daten zu sammeln. Für die hier entwickelte Sprachsteuerung sind es eine Menge an Daten von Sprachaufnahmen, sodass man am Ende eine aussagekräftige Grammatik erstellen kann, mit der das Spracherkennersystem trainiert werden kann. Dabei sollten möglichst alle Variationen der Sprache für die möglichen Befehle

im Experiment aufgenommen werden, damit möglichst *älles verstanden"* wird vom System. Dann lässt sich am Ende es auch programmiertechnisch umsetzen, dass mit einer Spracheingabe, auch die richtige Funktion ausgeführt wird.

Die WoO-Versuche werden aber meisten nicht von großen Firmen, wie Apple für ihre Sprachsteuerung *Siri* oder das aktuellste Beispiel von Google's Sprachsteuerung, genutzt. Meist einfach aus dem Grund der zu langen Entwicklungszeit und der Durchführung. Solche Unternehmen arbeiten auf Basis von vielen Daten die sie selber sammeln oder kaufen, was man **Big Data** nennt.

Big Data:

Beschreibt eine große Daten-Menge, die auf vielen unterschiedlichen Wege gesammelt und ausgewertet werden [Dum12]. Die Quellen der Daten können von jeglicher Kommunikationstechnologie kommen, von Firmen oder Behörden gesammelt werden, bis hin zu Überwachungssystemen und auch Bereiche, die bis dato für privat empfunden wurden. Der Drang von Behörden und großen Firmen, solche Daten für sich nutzen zu können, lässt die Art Daten zu sammeln immer mehr das Persönlichkeitsrecht verletzen.

2.2 Allgemeiner Aufbau von Wizard-of-Oz Versuchen

Bei WoO Versuchen gibt es mindestens 2 beteiligte Personen. Zum einen ist da der Proband, der das System testen soll und die ganze Zeit über in dem Glauben gelassen werden muss mit einer Maschine zu interagieren. Auf der anderen Seite ist da der *"Wizard"*. Der Wizard ist räumlich vom Probanden getrennt und dient dazu, das noch nicht existierende Programm im Hintergrund zu steuern. Er muss auf alle Eingaben des Probanden reagieren, wozu er sein eigenes Backoffice hat durch das sich alle Funktionen, sowie auch Textausgaben, steuern lassen. Wichtig ist dabei, das der Wizard ein konstantes Verhalten beibehält, auf die Eingaben bzw. Befehle des Benutzers immer gleich reagiert und auch möglichst für alle möglichen Eingaben des Testers auch eine Ausgabe bereit hält. Des weiteren gibt es noch verschiedene, häufig benutzte, Ausgaben mit Schnellzugriff (z.B. "Bitte warten...") um die Glaubwürdigkeit eines existierenden Systems zu wahren. Es gibt also den Probanden, der das System ausgiebig testen soll und mindestens einen Wizard, der die noch nicht vorhandenen Funktionen imitiert.

2.3 Allgemeiner Ablauf von Wizard-of-Oz Versuchen

Vor Beginn der eigentlichen Versuchsdurchführungen sollte man ein paar Testläufe machen, um sicher zu gehen, dass man als Wizard auf alle möglichen Eingaben reagieren kann, weil wie schon erwähnt, das Verhalten von Menschen unvorhersehbar ist und durchaus auch Anforderung ans System gestellt werden, die vorher so gar nicht zu erwarten waren. Außerdem dient es dem Wizard dazu sein Backoffice und die Abläufe des Experiments zu durchlaufen, um später schnell und richtig reagieren zu können. Wichtig beim Ablauf ist, das der Proband, wie schon erwähnt, zu keinem Zeitpunkt des Tests bemerkt, dass er nicht mit einem Computer arbeitet, sondern der Wizard die Funktionen ausführt. Dazu sollte der Wizard möglichst auf Fehler verzichten und sich

während des ganzen Ablaufes wie ein Computer verhalten, der aber alle Benutzereingaben gegenüber des Interfaces erkennt und umsetzt. Genau dies ist der entscheidende Punkt, warum solche Tests überhaupt durchgeführt werden. Der Wizard kann auf jegliche Eingabe angemessen reagieren, was ein Computer zur Zeit noch nicht kann. Da aber die Tester wissen, dass das Programm noch im Testlauf ist, sind kleinere Fehler die der Wizard macht, als normales Verhalten eines Computers zu betrachten. Doch durch die "Narrenfreiheit", die der Tester besitzt, was Befehle und Kommandos betrifft, ist es möglich das Verhalten von Benutzern eines Systems zu analysieren. Dadurch kann man die Nutzungsgewohnheiten von vielen Personen aufzeichnen und analysieren (z.B. ob kurze oder ausführliche Sprachkommandos genutzt werden) und daraus Rückschlüsse auf die weitere Entwicklung des Systems oder des Programms ziehen. Jedoch ist es ebenfalls wichtig eine Testperson nicht mehrfach testen zu lassen, sondern man muss versuchen ein großes Spektrum der späteren Zielgruppe als Tester einzusetzen. So lassen sich die Bedien-Gewohnheiten der späteren Kunden erfassen, analysieren und in die Benutzungsgestaltung der Oberfläche einarbeiten. Durch viele verschiedene Testpersonen ist außerdem gewährleistet, dass keine speziellen persönlichen Vorlieben eines Testers zu starken Einfluss auf die weitere Gestaltung haben. Dann wäre man sonst wieder dort, wo man angefangen hat: bei der Entscheidung über die weitere Entwicklung durch eine Person.

3 Bewertung der Validität

In der bisherigen Beschreibung der WoO Methode klang es so, das eine einmalige Durchführung ausreichen würde. Doch schon Kelley [Kel84] beschreibt, das die Schlüsselkomponente für einen validen Einsatz der WoO Methode einen mehrfachen Einsatz benötigt, es also kontinuierlich in den Entwicklungsprozess einbezogen werden muss. Mit voranschreitender Implementierung werden die *Eingriffe* des menschlichen Operanden abnehmen. Etwas neuere Kriterien zur Bewertung der Validität der WoO Methode stellten in jüngster Zeit auch Fraser und Gilbert(1991)[Fra91], Green, Hüttenrauch und Eklundh(2004)[Gre04], sowie Steinfeld, Jenkins und Scassellati(2009)[Ste09] auf.

3.1 Kriterien nach Fraser und Gilbert

Bei der Entwicklung ein computerbasierten Sprachsteuerung erarbeiteten Fraser und Gilbert drei wesentliche Anforderungen für eine valide Simulation in Verwendung der WoO Methode:

1. eine Simulation so *überzeugend* wie möglich zu gestalten, damit der Testkandidat zu keiner Zeit die Annahme hat, nicht ein computergesteuertes System zu bedienen und dementsprechend muss sich der Wizard verhalten.

2. das zukünftige System unter Berücksichtigung menschlicher Fähigkeiten simuliert werden muss.

3. es muss ebenfalls gewährleistet werden, dass das zukünftige System möglichst *genau* spezifiziert werden muss, das heißt das alle Funktionen genauestens festgelegt werden sollten, sie den Operateur bzw. Wizard bekannt sein sollten, damit er sich wie das zukünftige System verhalten kann und auch mögliche Fehler einbaut, die das System machen könnte

3.2 Kriterien nach Green u.a.

Die Autoren Green u.a. formulierten im Kontext der Nutzung eines Serviceroboters drei Kriterien unter Anwendung der WoO Methode unbedingt zu beachten sind. Im Detail benannten sie:

1. der Nutzer muss vor dem Start der Versuchsdurchlaufes möglichst genau vom Versuchsleiter instruiert werden, um die Frage zu klären "Was soll der Nutzer tun?"

2. es sollte vorher genau geschaut und überlegt werden, wie der Nutzer mit dem System interagiert um eine Handlungsweise zu erahnen, um somit die Frage zu klären "Was wird der Nutzer tun?"

3. und es muss das *Verhalten des Systems* (bzw. in diesem Fall des Roboters) geklärt werden, egal ob gesteuert durch den Operateur oder als autonomes System, um die Frage zu klären "Was soll das System tun?"

3.3 Kriterien nach Steinfeld u.a.

In dem Artikel beschreiben die Autoren zwar explizit Szenarien mit einem Roboter, doch lässt sich annehmen, dass die vorgestellten Szenarien auch im Kontext anderer technischer Systeme ihre Gültigkeit besitzen. Dem Aspekt Umwelt kommt hier in den WoO Versuchen eine große Bedeutung zu. Sie beschreiben unterschiedliche Szenarien, wo der Anteil der Menschlichen Interaktion und der Grad der Autonomie des Roboters jeweils unterschiedlich stark ist. Sie nehmen an das die Umwelt sowohl die Roboter-('Wizard') als auch die Menschen-('Oz') Komponente individuell beeinflusst, egal ob beide, oder sogar die Umwelt, real oder simuliert existieren. Im Einzelnen differenzieren sie zwischen sechs verschiedenen Szenarien (siehe Abbildung 1):

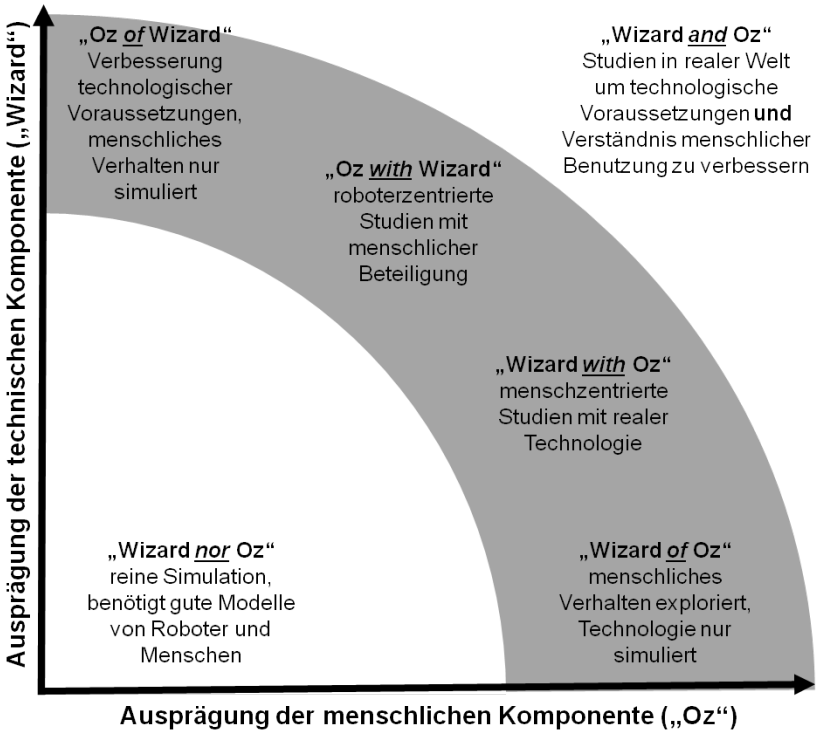


Abbildung 1: Mögliche Kombinationen von 'Wizard' und 'Oz'[wir13]

1. *'Wizard of Oz'* ist das klassische Modell in dem der Roboter durch den Menschen simuliert wird. Es wird lediglich der Einfluss des menschlichen Verhaltens ('Oz')in der jeweiligen Umwelt betrachtet, die Einflüsse des Roboters werden allerdings nicht erfasst. Es kann zwar ein realistisches Systemverhalten durch den Operateur angenommen werden, aber nicht realisiert werden bis das System nicht vollständig entwickelt ist.

2. *'Wizard with Oz'* hier liegt der Fokus immer noch stärker auf der menschlichen Komponente ('Oz'), wobei der Roboter('Wizard') nur begleitend betrachtet wird. Der Roboter ist hierbei aber schon ein tatsächlich implementierte Technologie.

3. *'Wizard and Oz'* in diesem Szenario wird sowohl der real existierende Roboter als auch der *'reale'* Mensch getestet in der realen angestrebten Umwelt, welches das wohl erstrebenswerteste Szenario ist in der Untersuchung der Mensch-Roboter-Interaktion. Dabei lässt sich das gesamte Verhaltensspektrum in der realen Umwelt betrachten.

4. *'Oz with Wizard'* legt den Fokus auf das Verhalten des Roboters, wobei die menschliche Komponente nur begleitend betrachtet wird, aber nicht explizit *'getestet'*.

5. *'Oz of Wizard'* der Fokus liegt auch hier ganz klar auf die Ent- und Weiterentwicklung des technischen Systems. Die menschliche Komponente wird dabei nur auf Grundlage von Modellen simuliert, welche auf experimentelle Verhaltensbefunde basieren.

6. *'Wizard nor Oz'* bei dem Szenario werden alle Bereiche simuliert, dies ist jedoch nicht sehr erstrebenswert.

4 Media Player

In der heutigen Zeit gibt es eine Vielzahl an unterschiedlichen Media Playern, mit vielen unterschiedlichen Funktionen, wo es meist keine großen Unterschiede in den Grundfunktionen gibt. Aber jeder hat sein eigenes Design und sie unterstützen unterschiedliche Dateiformate zusätzlich. Jeder Soft- bzw. Hardwareanbieter hat da seinen eigenen als Freeware direkt zur Verfügung stehenden Player. Im folgenden wird aber nur der WMP näher betrachtet, weil er der verwendete Player im Experiment ist und da das Ziel sein soll eine Musikbibliothek zu steuern, wird auch nur die Musikfunktionen betrachtet.

Windows Media Player

Der WMP ist der hauseigene Player von Microsoft in mittlerweile der zwölften Generation und ist mit dem Windows Betriebssystem meist vorinstalliert und damit einer der meist genutzten Media Player seiner Art. Bis zum WMP 9 gab es diesen sogar noch kostenlos für das Betriebssystem Mac OS X [Mic14]. Beim WMP gibt es 3 unterschiedliche Ansichten, um mit ihm zu arbeiten. Je nachdem in welcher Ansicht man den WMP schließt, in der öffnet er sich auch wieder bei erneutem Start. Zum einen wäre da die Bibliotheksansicht (siehe Abbildung 2), in der man, im Zentrum der Ansicht, eine Übersicht über die auf dem Rechner vorhanden Medien bekommt. In der linken Menüseite lässt sich die Darstellung im Zentrum nach Alben, Interpreten, Listen usw. sortieren. In der rechten Spalte kann man die ausgewählten Stücke sehen, die direkt als Liste gespeichert werden, auf einen Datenträger gebrannt oder durch Synchronisation auf andere Geräte geladen werden können. Die anderen Funktionen wären noch, die Suchfunktion, um gezielt nach Medien zu suchen, die Erstellung von Wiedergabelisten, entweder manuell oder automatisch nach Suchkriterien, das freigeben der Dateien im Stream oder die Organisation der Medienbibliothek, wo die Sortiereigenschaften verfeinert eingestellt werden können.

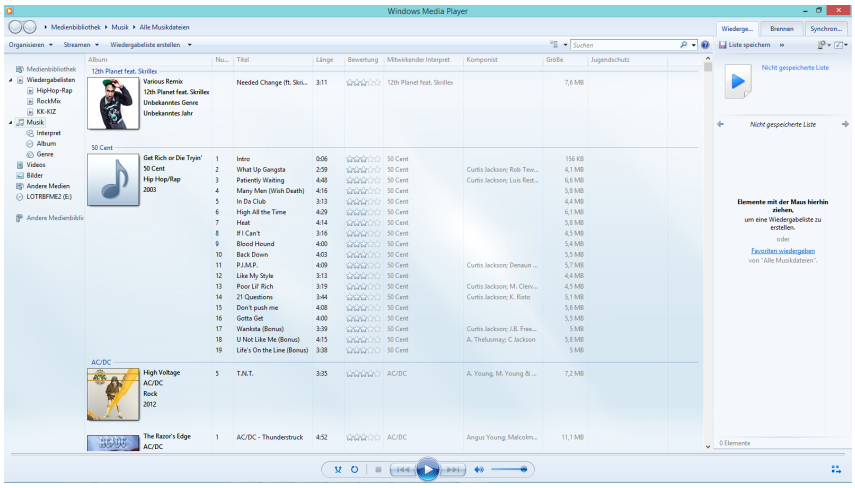


Abbildung 2: WMP Bibliotheksansicht

Die nächste Ansicht ist die, die sich während des Abspielens von Musik öffnet. Sollte jedoch keine Musik ausgewählt sein, wie in der Abbildung 3 zu sehen, kann man wählen

zur Bibliotheksansicht zu wechseln, um Musik zu wählen, einfach alle Titel anspielen oder das als letzte Gehörte abspielen. Mit einem Rechtsklick lassen sich zusätzliche Optionen, wie die zufällige Wiedergabe, einblenden der Liste oder Songs wiederholend abzuspielen, auswählen.

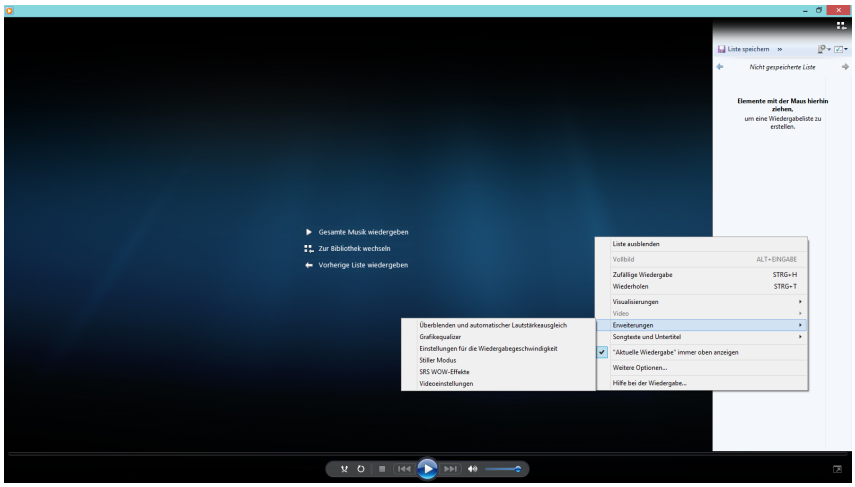
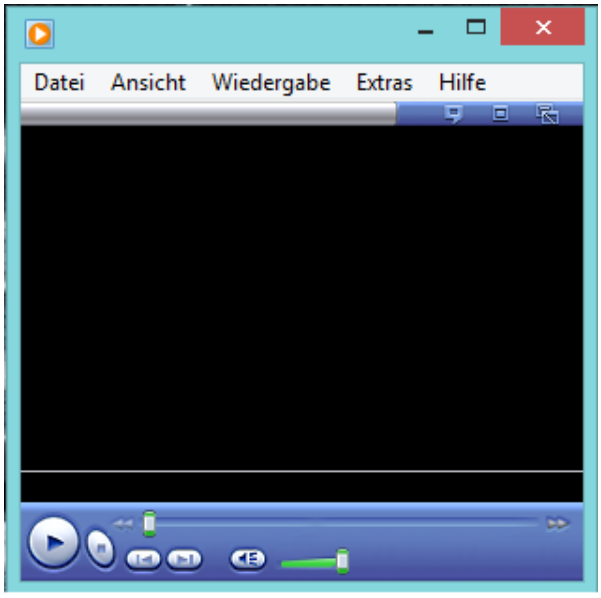


Abbildung 3: WMP Abspielansicht

Die letzte Ansicht ist die Designansicht, in der einfach nur die einfachen Bedienelemente (Play, Pause, Stop etc.) vorhanden sind, wie in allen Ansichten und der derzeit abgespielte Song angezeigt wird. Das interessante an dieser Ansicht ist nur, das man so seinen Player individualisieren kann, wie zum Beispiel im Batman Thema.



(a) Standartdesign

(b) Batmanthema

Abbildung 4: WMP Designdarstellungen

5 Experiment

Um den Spracherkenner später trainieren zu können ist es nötig eine Menge an Daten zu sammeln. Da aber noch kein existierendes reales System zur Verfügung steht, nutzen wir die schon vorher beschriebene WoO Methode, um das zukünftige System zu simulieren und Daten zu gewinnen. Nach Seinfelds Kriterien (siehe Abbildung 1) nutzen wir das klassische Modell der Durchführung *'Wizard of Oz'*.

5.1 Vorbereitung

Vor der Durchführung des Experimentes müssen aber einige Kriterien zuerst geklärt werden (siehe Kapitel 3), wo sich also folgende Fragen gestellt werden mussten:

1. Welche Funktionen soll *das System* erfüllen?
2. Welche Aufgaben soll *der Nutzer* erfüllen?
3. Was kann alles *adressiert werden* an Musikmedien?

Bei der Betrachtung der gängigen Media-Player fiel auf, das Funktionen wie Play, Stop oder Pause schnell zu bedienen Aktionen sind, die mit einem Mausklick oder den Knopfdruck auf einer Fernbedienung schnell erledigt sind. Deshalb sollten vorrangig Funktionen betrachtet werden, die häufig ausgeführt werden, aber einen Mehraufwand als einen Mausklick brauchen, die das Programm mittels Sprache steuern können soll. Die wichtigsten Aufgaben, die sich bei der Betrachtung herauskristallisierte waren:

1. Suche innerhalb der Medienbibliothek
2. Erstellen von Musiklisten
3. Bearbeiten der Listen
4. Organisieren der Musikbibliothek

Dabei sollte das System Musikdateien aber nicht nur nach den typischen Tags sortieren und aussuchen können, wie zum Beispiel Songtitel, Interpret oder Genre. Es soll zu dem auch möglich sein Titel nach der Stimmung oder ihrer Geschwindigkeit aussuchen zu können, sodass es möglich ist auch Suchanfragen zu stellen wie *'Computer, suche mir alle traurigen Lieder!'*. Damit ergibt sich für alle adressierbaren Attribute folgendes Entity-Relationship-Modell (ERM):

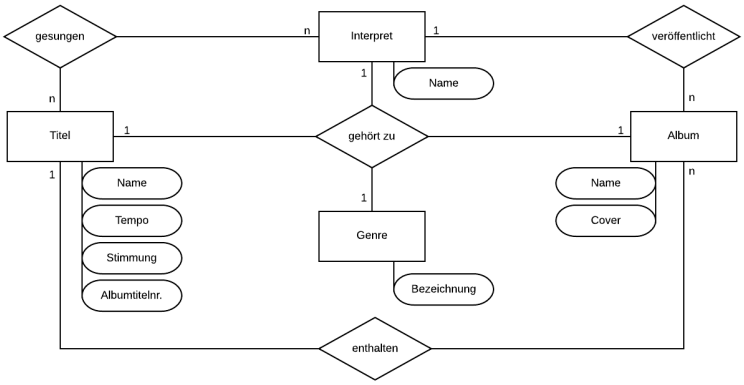


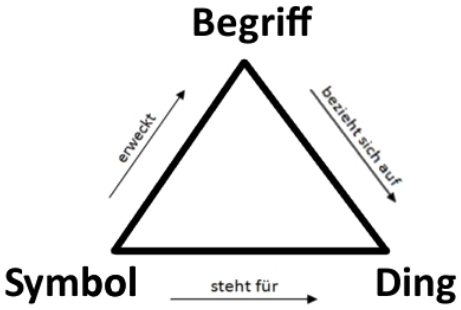
Abbildung 5: adressierbare Attribute

5.2 Entwicklung der Aufgabenstellung

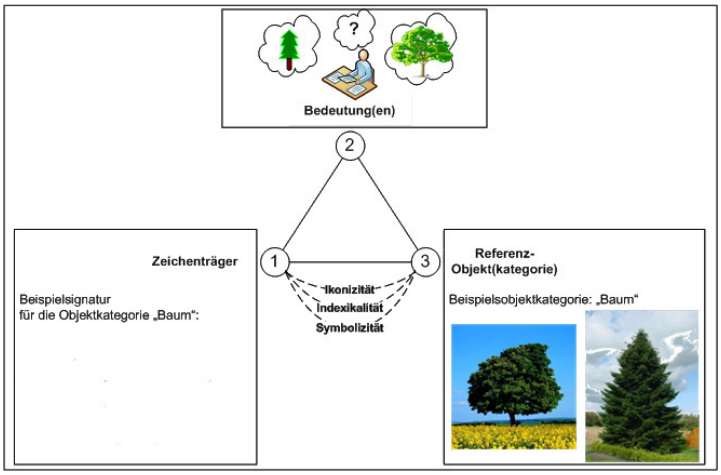
Bevor man dem Probanden das System testen lässt, muss ihm verdeutlicht werden, wie er mit dem System umzugehen hat und welche Funktionen er testen soll. Bei Dialogsystemen ist es dabei enorm wichtig, dem Probanden keine Worte in den Mund zu legen, die die Spracheingaben beeinflussen könnten und somit das Ergebnis am Ende verfälschen. Es gibt dabei zwei Varianten, um dem Probanden mit dem System interagieren zu lassen. Die eine Methode wäre, nach einer ausführlichen Einweisung ins System, ihn frei mit dem System agieren zu lassen, sodass er selbst entscheidet, wie und wann er die einzelnen Features testet. Das ist aber eher selten der Fall, aus dem ganz einfachen Grund, das damit nicht gewährleistet ist, das alles getestet wird, was an Möglichkeiten zur Verfügung steht und bei einer großen Funktionsfülle der *'Wizard'* schnell Fehler machen könnte all diese zu bedienen und somit die Simulation durchschaut wird. Die andere Möglichkeit ist, der Testperson die Aufgaben vorzugeben, welche auch in diesem Experiment angewandt wird. Hierbei sollte natürlich möglichst auf Text in der Aufgabenstellung verzichtet werden, um auch da nicht die Spracheingaben zu verfälschen und die größte mögliche Variation der Sprache zu bekommen. In den entwickelten Aufgabenstellung wurde deshalb versucht gänzlich auf Text zu verzichten, die einzige Ausnahme wurde bei Eigennamen, wie Bandnamen, vereinzelt Genre oder Listenbezeichnern gemacht, die nur als Platzhalter fungieren und variabel austauschbar sind. Die Schwierigkeit besteht darin, das nicht alle Bilder gleich gedeutet werden und überhaupt eine Möglichkeit zu finden etwas bildlich darzustellen.

5.2.1 Semiotisches Dreieck

Das simiotische Dreieck ist ein Modell der Sprachwissenschaft und Semiotik , welches verdeutlichen soll, dass sich ein Symbol nicht direkt auf ein Ding bezieht. Es ist viel mehr eine Vorstellung/Begriff dazu notwendig, die wir benötigen, um von einem Zeichenträger zu einem Gegenstand die Verbindung herzustellen.



(a) vereinfachtete Darstellung [20008]



(b) Am Beispiel Baum (bearbeitet) [Hru10]

Abbildung 6: Darstellung semiotisches Dreieck

Durch die Sprachnorm besteht eine Konvention, sodass jedes Mitglied einer Sprachgemeinschaft mit einem Wort ein Bild verbindet. [Lew94] Bereits in der antiken Sprachphilosophie wurde erkannt, dass Wörter nicht von Natur aus zu ihren Inhalten gehören, sondern durch Konvention. Um sozial handeln zu können, müssen sich Einzelne aber auch an diese Konventionen halten. Doch ist es so, dass nicht ein jeder mit dem selben textit'Symbol' das gleiche *'Ding'* verbindet, weil über die Ebene der Vorstellung im Kopf, sich ein anderes Bild bilden kann. Betrachten wir dies am Beispiel des Wortes Baum (siehe Abbildung 6b), so verbinden manche Menschen in ihrer Vorstellung damit einen Laub- und andere eben einen Nadelbaum.

Doch ist es in unserem Experiment nun so, dass wir das Ding mittels Bild vorgeben, damit die Probanden so auf die richtige Symbolik schließen und somit genau wissen, was für eine Aufgabe sie zu bewältigen haben. Die Problematik besteht, dass nicht einfach nur erreicht werden soll, dass die Tester einen Baum erkennen, sondern aus vielen Bildern den Zusammenhang erkennen und somit die Anforderung ans System stellen können.

5.2.2 Erste Entwicklungsschritte

Da am Lehrstuhl Kommunikationstechnik (KT) die WoO Methode noch nie in dieser Form durchgeführt wurde, mussten die Aufgabenstellungen ganz neu entwickelt werden. Angelehnt an die Durchführung einer früheren Versuchsreihe von M. Sc. Kathleen Karnagel [Kar12] wurde als Hintergrunddesign für die Aufgabenblätter die Ansicht eines einfachen übersichtlichen Media-Players gewählt, um an das Thema zu erinnern, das ein solcher bedient werden soll. In ihrer Thesis ging es um die Modellierung einer Home-Entertainment Steuerung in der bestimmte Filme gesucht und sortiert wurden und sie wählte als Hintergrund eine Filmklappendarstellung. Um mit Listen und einzelnen Titel zu arbeiten wurde angefangen mit unterschiedlichen Varianten zu arbeiten, um die Anforderung ans System zu verdeutlichen. Diese Bestanden meist nur aus einzelnen zusammengesetzten Bildern, doch waren alle Aufgabenstellungen unterschiedlich und weit

entfernt von einer konsistenten Darstellung. Betrachten wir beispielsweise Abbildung 7. Hier wurden mittels *'Pluszeichen'* einzelne Elemente verbunden, um einen Anforderung ans System darzustellen.

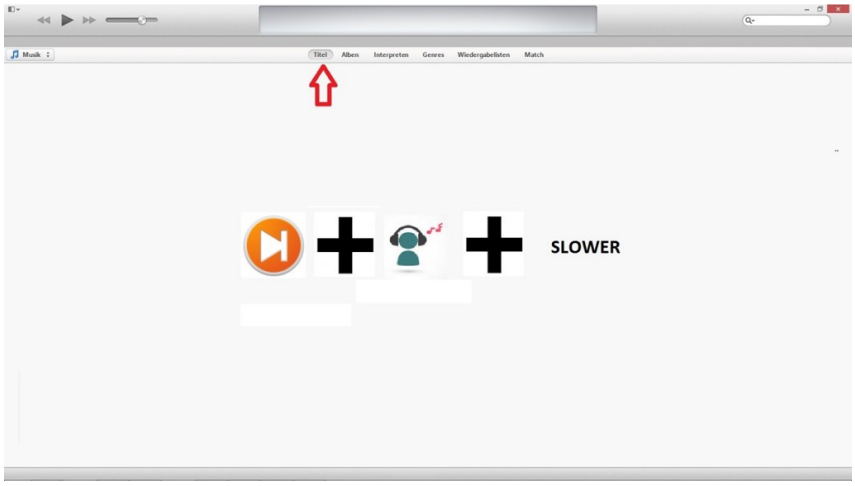


Abbildung 7: Bsp. eins erster Versuch

Die Schwächen in dieser Darstellung sind klar. Zum einen sind die Schaltzeichen eines Players nicht jedem geläufig und sollten deshalb in einen Zusammenhang mit den anderen Bedienelementen gebracht werden und auch wenn das geschriebene Wort in Englisch ist, könnte es den Tester beeinflussen, was ja anfangs der Grund war auf Textvorgaben zu verzichten.
Dieses Beispiel ist mit Absicht gewählt. Wenn man selber solche Aufgabenstellung entwickelt, dann kennt man natürlich die Intention hinter jedem Bild und worauf man hinaus will, aber da dies jeder verstehen soll, muss immer wieder getestet werden, wie die Darstellungen gedeutet werden, um ob sie wirklich erkannt und verstanden werden. Im Anhang kann man eine Auswahl der Erstversuche betrachten, wovon welche zwar schon die gestellte Aufgabe erkennen lassen, aber immer noch nicht Konsistent sind (siehe Anhang C). Es musste erreicht werden, eine einheitliche Darstellung zu finden, die für jede Aufgabenstellung nur geringfügig angepasst werden muss, doch womit sich alle möglichen Aufgabenstellung darstellen lassen. Wie bei vielen Entwicklungen ergeben sich aus den ersten fehlerhaften Versuchen gute Ideen für ihre Verbesserung und Erweiterung. Aus den im Anhang zu sehenden Versuchen ergaben sich zwei Varianten, die später für die Versuchsdurchführung genutzt wurden.

5.2.3 Aufgabenblattversion 1

Die erste Version der AGSt (siehe Abbildung 9) entwickelte sich aus dem eben gezeigten Erstversuch (siehe Abbildung 7). Aus den ersten Fehlversuchen wurde das System des syntagmatischen Zusammenhängens vom drei einzelnen Bildbereiche übernommen in dem die einzelnen Bilder paradigmatisch untereinander variabel Ausgewählt werden können.

syntagmatisch:

Syntagma ist ein Begriff der Linguistik eingeführt von *Ferdinand de Saussure* und bezeichnet eine Gruppe von Elementen in einer vorliegenden Äußerung [Kjo09]. Ein nicht notwendiges Kriterium ist, das die einzelnen Elemente nebeneinander stehen, doch wird es häufig so praktiziert, weil man, aufgrund des linearen Charakters von Sprache, die Verkettung von Einzelelementen auf der horizontalen Achse zu einer komplexeren Einheit beschreiben kann. Die Elemente werden durch Zergliederung (Segmentierung) gewonnen und können *'aus Lauten, Wörtern, Wortgruppen, Teilsätzen oder ganzen Sätzen bestehen.'* [KMB08]

paradigmatisch:

Der Gegenbegriff zum Syntagma ist das Paradigma. Roland Barthes machte in seinen Elementen der Semiologie (frz. 1965) diesen Zusammenhang für die Linguistik zentral, so dass man heute sagen kann: *'Paradigmen und Syntagmen bilden die Strukturmuster der Sprache.' 'Im Syntagma werden Elemente kombiniert, im Paradigma (virtuell) gegenübergestellt.'*[Koc10]

Das Verhältnis von Syntagmen und Paradigmen kann durch folgendes Modell dargestellt werden.

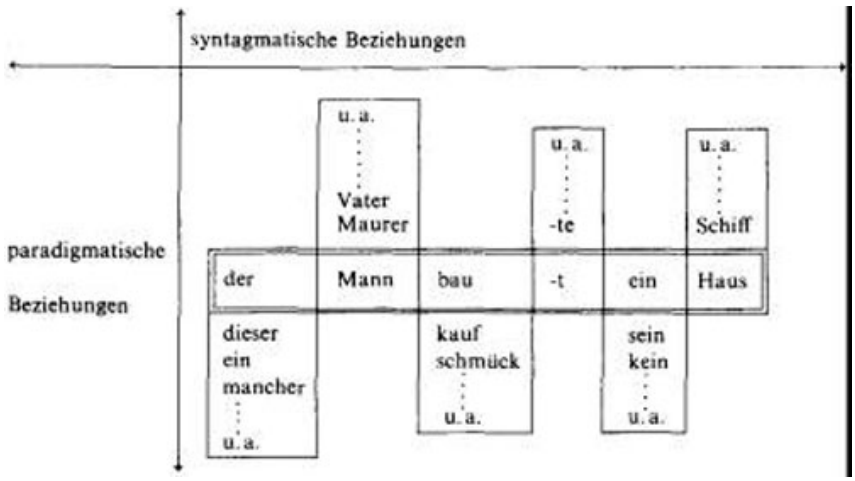


Abbildung 8: Modell Syntagma-Paradigma [Ulr72]

Sprachliche Elemente, die zusammen in einem Syntagma stehen können, stehen in syntagmatischer Beziehung zueinander. Sprachliche Elemente, die an derselben Stelle

eines Syntagmas eingesetzt werden können, stehen zueinander in einer paradigmatischen Beziehung.

So stehen in dieser AGSt die drei Bildbereiche in syntagmatischer Beziehung zueinander und die Bilder in den einzelnen Bereichen, in einem paradigmatsichen Zusammenhang. So lassen sich Play, Pause oder Suchen variabel ersetzen. Welches Element im einzelnen Bereich ausgewählt ist, wurde mittels roten Rahmen gekennzeichnet und alle markierten Bilder zusammen ergeben letztendlich die zu stellende Anforderung. In diesem Beispiel wäre die Aufforderung ans System zum Beispiel *'Computer, spiele mir eine Liste von schnellen Liedern!'*, was natürlich nur eine Variante wäre, das System anzusprechen.

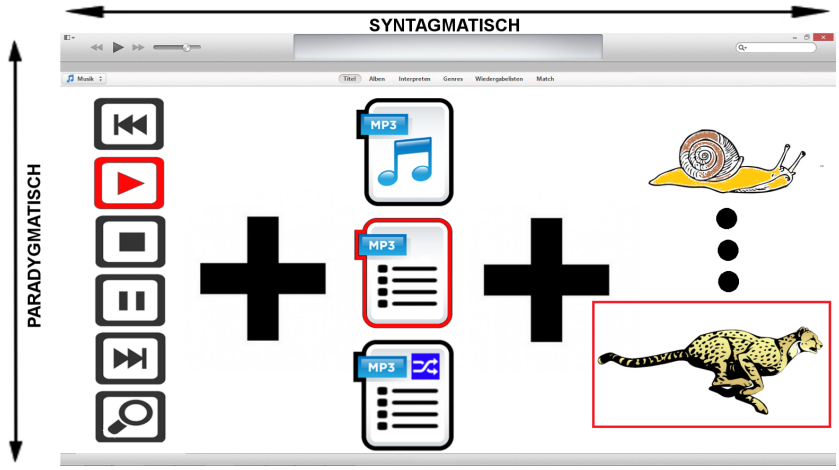


Abbildung 9: AGSt Variante 1

Im ersten Bereich dieser Aufgabenstellungen sind dabei immer alle zur Verfügung stehenden Funktionszeichen, wie Play-, Pause-, Stop- oder die Suchfunktion, die für die Aufgabenerfüllung nicht unbedingt alle gebraucht werden, doch für Konsistenz sorgen. Denn alle Schaltzeichen in Verbindung erinnern mehr an die Bedienelemente eines Players. Die Objekte die man bei einem Musikplayer nutzt, sind entweder einzelne Titel oder mehrere in Form einer Liste und da wir den Abspielmodus des zufälligen Wiedergebens einer Liste auch nutzen wollen, gibt es dafür noch mal eine zusätzliche bildliche Darstellung im mittleren Bereich der AGSt. Die beiden Teile der Aufgabenstellung sind auch immer gleich bei dieser Art und Weise, die Aufgabe zu stellen. Der Bereich auf der rechten Seite hingegen variiert zwischen den einzelnen Aufgaben. Er stellt jedoch immer etwas bildlich da, was den Titel näher beschreibt. Dies kann die auszuwählende Musik nach Geschwindigkeit, Genre, Stimmung usw. genauer bestimmen. Dieser Teil kann aber auch weggelassen werden, wenn es nicht notwendig ist die gewählte Musik näher zu bestimmen, wie in zwei Beispielen zu sehen ist (siehe Anhang D)

5.2.4 Aufgabenblattversion 2

Die zweite Variante der Aufgabenstellung, entstand aus dem Grund, das mit der Ersten sich zwar alle normalen Funktionen eines Musikplayers umsetzen ließen, aber es sollte ja auch mit der Sprachsteuerung umgesetzt werden, die Musikdaten bzw. Listen zu organisieren. Angelehnt ist diese AGSt an eine Theorie von Burrhus Frederic Skinner. Er war ein US-amerikanischer Psychologe der Verhaltensstudien an Tieren gemacht hat und diese auf den Menschen übertragen hat und damit einer der prominentesten Vertreter des Behaviorismus wurde. In seinem Buch *'Verbal Behavior'* wendet er die in Experimenten an Tieren und Menschen gewonnenen Erkenntnisse der Verhaltensanalyse auf das sprachliche Verhalten an. [Ski91]

ABC-Schema:

Verhalten unterliegt aus verhaltensanalytischer Sicht der Kontrolle durch die Umwelt, oder anders formuliert: Verhalten wird von der (gegenwärtigen und vergangenen) Umwelt des Organismus geformt. Dabei ist zu unterscheiden zwischen der

Stimuluskontrolle – Womit der Einfluss aller Ereignisse in der Umwelt des Organismus gemeint ist, die zeitlich vor dem Verhalten stattfinden, also der vorausgehenden Bedingungen des Verhaltens (Antecedents = A)

und der

Verstärkerkontrolle – Damit ist der Einfluss der dem Verhalten zeitlich nachfolgenden Umweltereignisse gemeint, also der Konsequenzen des Verhaltens (Consequences = C).

Mit Hilfe des ABC-Modells (das „B“ steht für Behavior, also Verhalten) kann die Form und die Häufigkeit von Verhalten erklärt werden:

$A \rightarrow B \rightarrow C$

Man spricht hier auch von der Dreifachkontingenz (three-term-contingency) des Verhaltens. Vereinfacht lässt sich die Kernaussage dieses Modells so umschreiben: Verhalten wird von vorausgehenden Bedingungen (der Situation, den Stimuli) ausgelöst und von Konsequenzen (Verstärkern) aufrechterhalten.

Es ist vielleicht etwas weiter gegriffen, als es Burrhus F. Skinner in seinem Modell des ABC-Schemas beschreibt, doch wollen wir mit dieser Aufgabenstellung durchaus das Verhalten der Versuchsperson kontrollieren, damit er die richtige gewollte Anforderung ans System stellt.

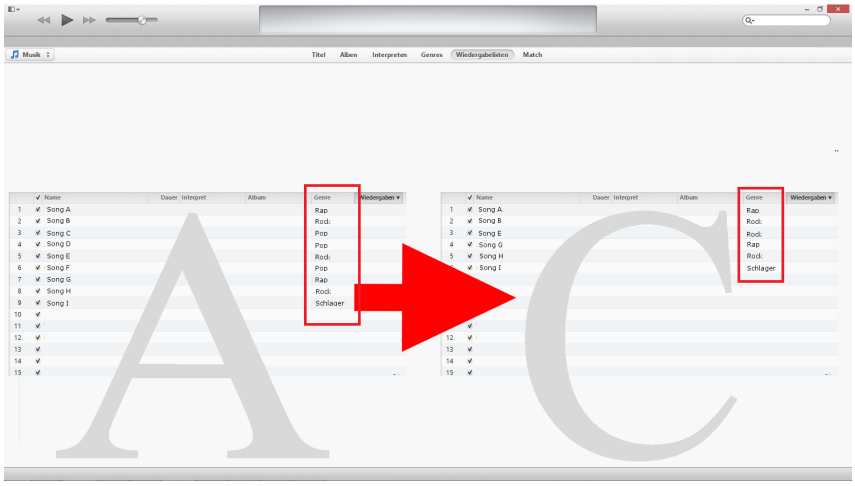


Abbildung 10: AGSt Variante 2

In unserem Beispiel sind immer zwei Listen gegeben und optional auch der Listenname, der gegebenenfalls vorgegeben ist (siehe Anhang D). Die linke Liste ist die bereits vorhandene und zu bearbeitende. Sie existiert schon vor dem geplanten Verhalten der Testperson und ist damit unser *Antecedents*. Die andere Liste, auf der rechten Seite der AGSt, zeigt den Zustand der Liste nachdem sie bearbeitet wurde. Somit ist sie die *Konsequenz / Consequences* aus dem Verhalten das wir durch den Probanden erwarten (siehe Abbildung 10). Allein durch diese beiden Vorgaben sollte es uns möglich sein, das gewünschte Verhalten beim Tester hervorzurufen und somit sein Verhalten vorab zu kontrollieren. In diesem Beispiel haben wir in der Liste des Antecedents erst mehr Titel als in der Konsequenz, woraus als Verhalten resultieren muss, dass als Anforderung ans System etwas kommt, wie zum Beispiel *'Computer, lösche alle Titel des Genre Pops!'*

5.3 Aufbau des Experimentes

Der Aufbau des Experimentes teilt sich in zwei Bestandteile. Zum einen den räumlichen Aufbau im Labor des Lehrstuhls KT, wo nochmal einzeln auf die Position und den Aufbau beim Versuchsleiter und auf die des Probanden eingegangen wird. Der andere Abschnitt befasst sich dann nochmal mit dem rein technischen Aufbau, damit für kommende Versuche gewährleistet ist, dass der Aufbau genau gleich rekonstruiert werden kann.

Räumlicher Aufbau Sprachlabor:

Das Labor ist in zwei Räume aufgeteilt, die komplett schallisoliert voneinander getrennt sind (siehe Abbildung 15). So sind, wie in Kapitel 2 schon beschrieben, beide beteiligten Personen komplett räumlich getrennt. Der Versuchsleiter (*'Wizard'*) sitzt dabei in der Aufnahmekabine und der Proband steht, mit den Augen auf den großen LCD Bildschirm gerichtet, zentral im großen Raum des Sprachlabors.

Abbildung 11: Raumansicht Sprachlabor

1. *'Wizard' (Versuchsleiter)*: Er sitzt in der Aufnahmekammer und hat auf dem Bildschirm die Möglichkeit, sowohl seinen eigenen Bildschirm zu betrachten, wo neben dem Aufnahmeprogramm Adobe Audition auch die Reihenfolge der Aufgabenblätter geöffnet ist, als auch den Bildschirm des Versuchskandidatens, auf dem der Mediaplayer vom Operateur gesteuert wird. Er kann über die Kopfhörer alle Spracheingaben des Testers und auch die angespielte Musik mithören und wenn nötig über das Mikrofon mit dem Probanden in Sprachkontakt treten. Um sein eingerichtetes Backoffice Fehlerfrei steuern zu können, liegt beim Arbeitsplatz des Versuchsleiters eine Übersicht über alle mögliche Kurzwahl-Tastenkombinationen, die er steuern kann. (siehe Anhang F) Die im Anhang befindliche Übersicht wurde während des Experimentes noch um weitere Auswahlmöglichkeiten erweitert, da der Operateur auf weitere unerwartete Anfragen der Tester reagieren musste.

2. *'Oz' (Testperson)*: Die Testperson steht vor einem Mikrofon und trägt ebenfalls Kopfhörer, falls der Wizard ins Geschehen eingreifen muss. Auf dem LCD Display sieht er nur, wie der Mediaplayer scheinbar automatisch auf die Eingaben reagiert. Die Arbeitsblätter kann er nacheinander auf einem im zur Verfügung gestellten Tablet-PC betrachten.

Technischer Aufbau Sprachlabor:

Aus dem Absatz zum räumlichen Aufbau ist bekannt, das der große liquid-crystal Display (LCD)-Monitor dazu diente, dass die Versuchskandidaten das Verhalten des Windows Media Players beobachten konnten. Damit der Operateur in der Aufnahmekammer auch Zugriff auf die Ansicht des LCD-Monitors bekommen konnte, wurde mittels Ultra Virtual Network Computing (UVNC) Viewer auf das Bild zugegriffen. Um genau zu sein, ist das Bild des LCD-Monitors das gespiegelte Bild von Monitor 1 (siehe Abbildung 12). Um einen Bereich zu haben, der für die Testperson unsichtbar ist, wo aber die zusätzlich gebrauchten Programme des *Wizards* laufen, wurde mit dem Monitor 2 die Desktop-Fläche um einen weiteren Bildschirm erweitert. So hat der Operateur die Sicht auf 2 Monitore, die aber nur mittels Schiebebalken beide zu betrachten sind. Dazu muss erwähnt werden, dass nicht jedes Viewer-Programm die Sicht auf mehr als einen Bildschirm zulässt, das heißt, wenn dies benötigt wird, sollte man wirklich UVNC nutzen.

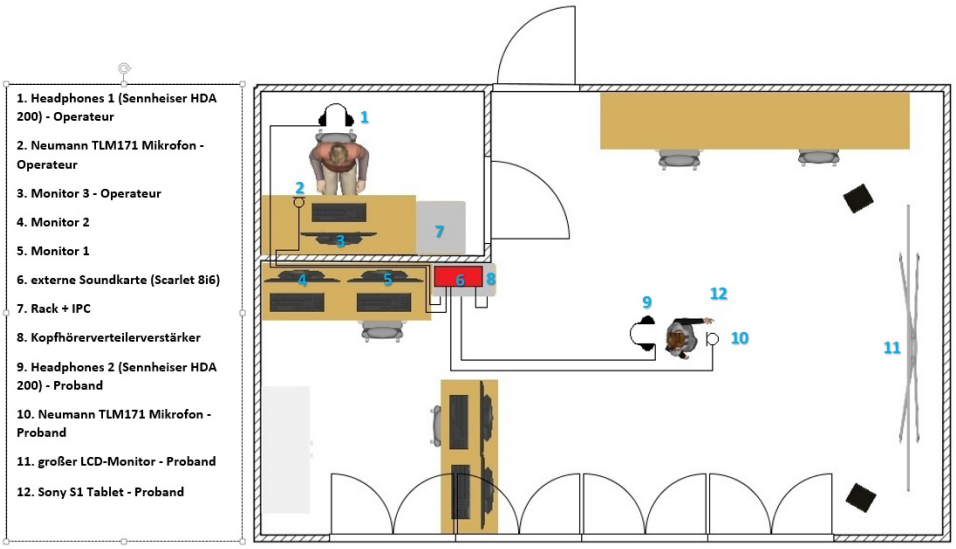


Abbildung 12: technischer Grundriss Sprachlabor

Beide Seiten des Experimentes nutzten ein Neumann TLM 171 zur Aufnahme der Sprache und damit der Proband seine vermeintlichen Sprachanweisungen ans System machen kann. Es wurden Sennheiser HDA 200 Kopfhörer verwendet, um die Musikwiedergabe und die mögliche Kommunikation zu hören. Zusätzlich sollte mit einem Headset aufgenommen werden, doch lies das die externe Soundkarte (Scarlet 8i6), an der die Neumann Mikrofone angeschlossen waren, mit der Phantomspeisung von nicht zu. Die Phantomspeisung dient dazu die Kopfhörer mit Strom zu versorgen, doch ist sie für die einfachen Headsets geringer als bei den HDA 200. In Abbildung 13 kann man detailliert sehen, wo die Kopfhörer am Kopfhörerverteilerverstärker und Mikrofone an der Scarlet 8i6 angeschlossen wurden und diese miteinander verbunden sind. Für den Testkandidaten bestand sogar die Möglichkeit zwischen 2 Headset zu wählen, denn neben dem HDA 200 Sennheiser Kopfhörern, gab es noch ein einfaches Headset von Sennheiser, das angeschlossen war. Zur Regelung der Pegel wurde die Scarlett Mixcontrol genutzt mit der der Operateur auch die Möglichkeit besaß sich zu *muten*, um mögliche Eingabegeräusche der Tastatur nicht zum Probanden durchdringen zu lassen und somit die Simulation auffliegen zu lassen. (Den technischen Schaltplan zum Aufbau findet man im Anhang G.)

Abbildung 13: Anschlüsse Soundkarte / Kopfhörerverteilverstärker

Alle genutzten Programme liefen ausschließlich über den Rechner (IPC + Rack) in der Aufnahmekammer. Außer dem Windows Media Player und der Scarlet Mixcontrol, wurde wie schon erwähnt Adobe Audition zur Aufnahme aller Spracheneingaben genutzt. Das aber wichtigste Programm für das Backoffice des *'Wizard'* war AutoHotkey. Mit AutoHotkey lassen sich mittels selbst festgelegten Schnellwahltasten viele Funktionen auslösen. Es lassen sich nicht nur bestimmte Dateien öffnen, Programme starten oder gar schließen, sondern es ist auch möglich mit diesem Programm ganze Eingaben der Tastatur machen zu lassen, die Bewegung der Maus zu kontrollieren und eigene Scriptfunktionen schreiben. Das Autohotkey Script läuft ungesehen im Hintergrund und blieb somit vor dem Probanden verborgen.

Hinweis: Auch wenn AutoHotkey sich wunderbar eignet solch ein Backoffice umzusetzen, machte es doch manchmal Probleme bei dem verwendeten Aufbau mit UVNC. Vorprogrammierte Tastatureingaben wurden manchmal nicht ausgeführt und sollte vor dem Start des Experiments überprüft werden. Dies geschah aber nur beim erneuten Start von AutoHotkey und nicht während des Versuchsablaufs.

5.4 Systemausgabe

Während des Experimentes muss der Proband die ganze Zeit in dem Glauben gelassen werden mit einem technischen System zu interagieren. Hierfür muss der Versuchsleiter die Systemausgaben ausreichend vorbereiten. Auf jede Eingabe des Testers muss während des gesamten Experimentes eine schnelle Antwort erfolgen und dabei muss es wirken, dass alle Eingaben auch vom System wirklich verstanden werden. Hierfür wurden, zu den musikalischen Ausgaben in der Musikdatenbank, auch Fehlermeldungen vorbereitet. Gebraucht werden in jedem Fall Standartausgaben, um auch auf unerwartete Eingaben reagieren zu können. Diese sind zum Beispiel:

1. *Das System hat die Eingabe nicht verstanden, bitte wiederholen Sie Ihre Eingabe!*Diese Nachricht ist besonders wichtig, um bei auftretenden Problemen auch kurzfristig Zeit zu gewinnen.

2. *Das System hat keinen Eintrag gefunden, bitte machen Sie eine andere Eingabe!* Diese Ausgabe erfolgte, wenn die Aufgabenstellung falsch ausgeführt wurde. Der Proband setzte dann mit dem nächsten Blatt fort.

Weitere Textausgaben dienten dazu, der Versuchsperson zu verdeutlichen, das Informationen zu Interpreten, Genre oder anderen Attributen fehlen, die das System zu einer erfolgreichen Bearbeitung der Anfrage benötigt.

Alle Systemausgaben mittels Text waren vorgeschrieben und konnten, wie alle anderen Funktionen, durch Tastenkombinationen, mit einem Druck ausgelöst werden. Direktes schriftliches antworten mittels Kommunikationsprogramm würden zu lange dauern und der Operateur könnte mit Rechtschreibfehlern den Eindruck verstärken, das es kein automatisch agierendes System ist. Die Textausgaben erfolgten über Pop-Up-Fenster, die erschienen, wie Beispielsweise in Abbildung 14.

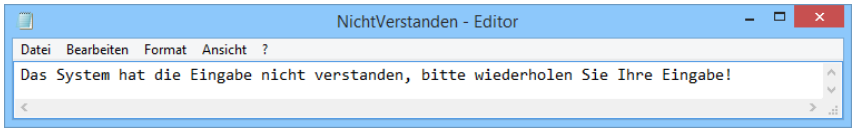


Abbildung 14: Textausgabe Fehlermeldung]

Die musikalischen Systemausgaben erfolgten über den in Kapitel 4 beschriebenen WMP. Es wurden hierfür nur die Bibliotheksansicht (siehe Abbildung 2) verwendet, um Suchanfragen durchzuführen und die Abspielansicht (siehe Abbildung 3), um die angespielte Musik darzustellen und wie die daran enthalten Listen bearbeitet werden.

5.5 Musik Datenbank

Musikdaten sind immer gekennzeichnet durch ihre Tags, wie zum Beispiel Interpret, Genre oder auf welchem Album sie zu finden sind. Für das geplante System sollten alle Titel zusätzlich Hinweise zur ihrer Stimmung und Geschwindigkeit haben. Viele Titel haben diese ganzen Tags jedoch meist nicht und oft sind nur die nötigsten Informationen von einem Titel bekannt, wie der Name und der Interpret. Um zu gewährleisten das die Datenbank unser Musikbibliothek zu den Titeln alle wichtigen Informationen hat, soll das System solche Informationen aus einer Tag Cloud beziehen. Diese besitzen für die meisten Titel alle möglichen Tags und werden von der jeweiligen Community und den Erstellern der Cloud auf dem neusten Stand gehalten. Mögliche Anbieter solcher Clouds findet man in der Wiki des Lehrstuhls Kommunikationstechnik unter dem Thema *'BA Mike Stöhr'*. Man kann sich entweder einen Datensatz, den man nutzen möchte, mittels XML selber erstellen oder die vollständige Datenbasis herunterladen. Mittels XML-Anfrage kann man dann auch die gewollten Tags abrufen.

```
warp@bullet-bill:~/tags$ curl --header 'Content-Type: application/xml; charset=UTF-8' \
> --silent --digest --user example:password --data-binary @body.xml \
> 'http://test.musicbrainz.org/ws/2/tag?client=example-0.3.7' | xmlindent
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://musicbrainz.org/ns/mmd-2.0#">
    <message>
        <text>OK</text>
    </message>
</metadata>
warp@bullet-bill:~/tags$
```

Abbildung 15: XML-Anfrage Datenbank

Weil die programmiertechnische Umsetzung nicht mehr Teil dieser Bachelorarbeit ist und noch keine Datenbank besteht, wurde in dieser Arbeit auf eine kleine ausgewählte Musikbibliothek zurückgegriffen. Welche Alben und Titel da im einzelnen enthalten sind, ist für die Arbeit weniger relevant, kann aber der Lehrstuhl-Wiki des Lehrstuhls KT entnommen werden. Es wurde aber ausschließlich eine legale Sammlung erworbener Musik für die Datenbank verwendet.

5.6 Zielgruppe

Für ein repräsentatives Aussage über das Experiment ist eine gewisse Anzahl an Teilnehmern verschiedener Altersklassen notwendig. In unserem Fall würde das alle Zielgruppen, d.h. Kinder, Jugendliche, Erwachsene und Senioren betreffen, da immer früher der Mensch an die Technik geführt wird und jeder der Musik hört in jedem Alter mit so einem Player zwangsläufig in Kontakt kommt. Es wäre erstrebenswert mindestens 20 Versucher pro Zielgruppe zu erhalten, doch würde dies den Rahmen dieser Bachelorarbeit sprengen.

Es wäre sicherlich zusätzlich Interessant, den Umgang mit dem System geschlechtsspezifisch zu betrachten. Eine zweite interessante Unterteilung wäre nach der technischen Affinität der Kandidaten, weil der unterschiedlich häufige Umgang mit technischen Geräten, besonders im Bereich Musik, uns vielleicht auch Unterschiede im Umgang mit dem geplanten System aufzeigen könnte. Dabei wäre besonders informativ die Seite der weniger technisch begeisterten Menschen, weil aus ihren gewonnen Daten das System für jedermann zugänglich gemacht werden könnte.

5.7 Durchführung

Vor der eigentlichen Durchführung standen erst ein paar Testdurchläufe mit Angestellten des Lehrstuhls KT, um zu gucken ob alle Aufgabenstellungen soweit verständlich sind und um den *'Wizard'* auf seine bevorstehenden Aufgaben in einem vollem Durchlauf zu trainieren. Als alle Abläufe funktionierten und der Versuchsleiter soweit trainiert war, das keine größeren Fehler auftreten, fanden die richtigen Versuchsläufe statt.

Alle Experimente fanden an sieben nicht aufeinanderfolgenden Tagen statt. Zur Verfügung stellten sich vier Frauen und zehn Männer, die alle in einem Alter zwischen 18 und 35 Jahren waren. Für ihren Versuch brauchten alle der daran teilgenommenen Personen ca. 35 min.

Vor der Durchführung musste jeder Testkandidat das vom Operateur erstellte Handout durchlesen, um Informationen zum Ablauf des Versuches und die Art und Weise wie die Aufgaben gestellt werden zu erhalten (siehe Anhang B). Außerdem mussten alle Teilnehmer eine Einverständniserklärung unterschreiben, damit die Daten jetzt und zukünftig für die Entwicklungen am Lehrstuhl genutzt werden dürfen. Danach folgt die Einweisung zur Bedienung des Tablets, um die Aufgabenstellung durchzublättern, die Einstellung des Mikrofons und die Unterweisung richtig ins Mikrofon zu sprechen. Zu dem wurde eine bildliche Darstellung vorher mit den Testern besprochen, weil diese nicht leicht aus dem Kontext zu verstehen ist. Dabei wurde aber darauf geachtet, das der Proband selber auf die Lösung kommt und ihm nicht etwas vorgesagt wurde, was er dann so im Versuch nachspricht.

Um zu testen, das alles verstanden wurde, wurden zum Anfang zwei Aufgabenblätter zusammen mit dem Probanden durchgegangen, wo jeweils eine Variante der AGSt getestet wurde. Diese wurden auch schon vom Versuchsleiter aufgezeichnet. Nach der Bearbeitung der beiden Aufgaben folgt eine kurze Pause. Sollten sich bei den beiden Testblättern irgendwelche Fehler in der Ausführung oder an der Tonaufnahmeaufzeichnung

gezeigt haben, konnte der Versuchsleiter nochmal ein Feedback geben für eine reibungslose Bewältigung der restlichen Blätter. Die nächsten 12 Blätter wurden dann selbstständig, ohne Eingriff des Versuchsleiters, vom Probanden bewältigt. Wenn der Kandidat ein Blatt nicht bewältigen kann und nicht zum richtigen Ergebnis gelangt, bleibt dem Operanden eine große Auswahl an Antwortmöglichkeiten durch Fehlermeldungen oder anspielen von der falschen Auswahl, falls in seinem Backoffice dafür eine Vorgabe vorhanden ist. Da im Vorgespräch der Proband darauf vorbereitet wurde, das auch Fehlermeldungen als Ziel mancher Aufgabenstellungen kommen, weil auch dieser Teil des Systems getestet werden muss, setzten manche Probanden mit dem nächsten Blatt fort. Der andere Teil merkte meist, das er in der Situation seine Anweisung unverständlich gestellt hat und formulierte dann nochmal eine neue Anforderung ans System. Trotz der Vorbesprechung und der erfolgreichen Testläufe mit den 2 Aufgabenblättern, kamen nicht alle Probanden bei jedem Blatt zu einem positiven Ergebnis. Es brach aber keiner den Versuch vorzeitig ab.

Nachdem alle 14 Aufgabenblätter vom Probanden bearbeitet wurden gab es eine Besprechung mit allen Probanden, wo sie zum einen über den Hintergrund dieses Experimentes aufgeklärt wurden und zum anderen ein Fragebogen ausfüllen mussten (siehe Anhang A). Der Fragebogen diente dazu, sowohl das Szenario als auch die erstellen Aufgabenblätter zu bewerten.

6 Bewertung und Auswertung

6.1 Stärken und Schwächen

Stärken: Schnelligkeit und Kosteneffizienz

Die Stärke des Wizard of Oz Szenarios liegt klar darin, dass sie eine sehr schnelle Form des Prototyping darstellt, da die zu testenden Funktionen nicht aufwendig implementiert werden müssen, sondern deren Simulation durch einen menschlichen Operateur, den Wizard, erfolgt. Dies spart Zeit und Kosten, da ein System bereits vor seiner Implementierung geprüft werden kann und ein aufwendiger Abbau und anschließend aufgrund etwaiger Mängel nötiger Wiederaufbau umgangen werden kann. Zu dem lassen sich weiter Informationen aus solchen Experimenten gewinnen und man erhält vorab ein Feedback, wie sinnvoll einige der Applikationen für das Gesamtsystem sind. Der letzte positive Punkt und der wichtigste für die Universität ist, die Sammlung an Daten, um später den Spracherkenner zu trainieren zu können.

Schwächen: ethische, methodische und technische Probleme

In Anlehnung an Riek [Rie12] lassen sich aus den berichteten Befunden drei wesentliche Probleme von Wizard of Oz Szenarien finden. Einen wesentlichen Aspekt bildet der ethische Bereich. Da die Versuchsperson während des Ablaufes im Glauben gelassen wird, mit einem technischen System zu interagieren, die Interaktion tatsächlich jedoch mit einem menschlichen Gegenüber erfolgt, werden die Probanden eigentlich getäuscht. Deshalb ist ein ausführliches Debriefing nach der Durchführung unbedingt notwendig, um den Tester über all das aufzuklären. Methodische Probleme können darin liegen, dass es dem Wizard schwer fallen kann, vorhandene Möglichkeiten und Grenzen des späteren Systems konsistent zu *'spielen'* und dass gerade bei Systemen mit vielen Funktionen, der Operateur bei zu vielen Aufgaben unter Druck Fehler begeht. Dieser Problematik kann aber durch ein gutes und intensives Training vorab sowie den Einsatz mehrerer Wizards begegnet werden. Des weiteren sind Wizard-of-Oz-Untersuchungen meist experimentell ausgerichtet und finden daher in einem kontrollierten Untersuchungskontext statt, weshalb es sich hier um eine unnatürliche Situation handelt. Auch findet hier keine wirkliche Interaktion zwischen Mensch und Technik, sondern zwischen Mensch und Mensch statt, was bei der Interpretation der Befunde stets berücksichtigt werden sollte. Von technischer Perspektive aus äußern manche Autoren schließlich Bedenken, dass dem System, durch die Simulation, Funktionen *'aufgezwungen'* werden, die es im implementierten Zustand möglicherweise gar nicht leisten kann und kommen daher zu der Empfehlung, anstatt der Simulation lieber tatsächlich existierende Systeme zu untersuchen. Dieser Problematik kann jedoch durch eine möglichst genaue Beschreibung des Systems vorab, wo klar die Grenzen festgelegt sind, entgegengewirkt werden.

Zusammenfassend kann man also sagen, das die Nutzung der WoO-Methode für kleinere Forschungen innerhalb von Universitäten, die nicht auf Big Data zugreifen können, wie große Konzerne, die beste Art und Weise sind, sich Daten zu sammeln und somit ihre Systeme zu entwickeln, doch immer berücksichtigt werden sollt, das dies unter nicht realen Bedingungen erfolgt.

6.2 Auswertung vom Fragebogen

Das abschließende Debriefing hat gezeigt, dass das simulierte System sehr gut gearbeitet hat. So fand nur ein Proband es so perfekt, dass er dahinter eine *'zusätzliche'* menschliche Steuerung vermutete. Zwei der am Experiment teilgenommenen Tester wollten die Sprachsteuerung sogar für ihren privaten Mediaplayer verwenden. Bei den Ausgaben des Systems, kam genau das als Ergebnis raus, was die Probanden erwartet hätten nach ihrer Spracheingabe, auch wenn sich nicht alle mit jedem der erfragten Genre auf den Aufgabenblättern auskannten. Doch konnten alle mit den gewählten Beispielen in der Datenbank etwas anfangen.

Die Aufgabenblätter wurden zum größten Teil von allen Teilnehmern verstanden und auch erfolgreich bewältigt, was auch damit zusammen hängt, das sie vorher gut vorbereitet wurden durch das Handout. Doch es fiel auf, das ein Aufgabenblatt den meisten anfangs Probleme bereitet hat. In den Vorversuchen fiel nur einmal auf, dass das Aufgabenblatt Probleme macht. Abhilfe wurde aber dadurch geschaffen, das Aufgabenblatt so einzureihen, dass die Probanden aus den vorhergehenden Blättern lernen, die Bedeutung des dritten Bereiches auf der ersten Aufgabenblattvariante zu verstehen. Trotzdem hatten 60 Prozent der Versuchsteilnehmer das folgende Aufgabenblatt nicht auf Anhieb erfolgreich bewältigt.

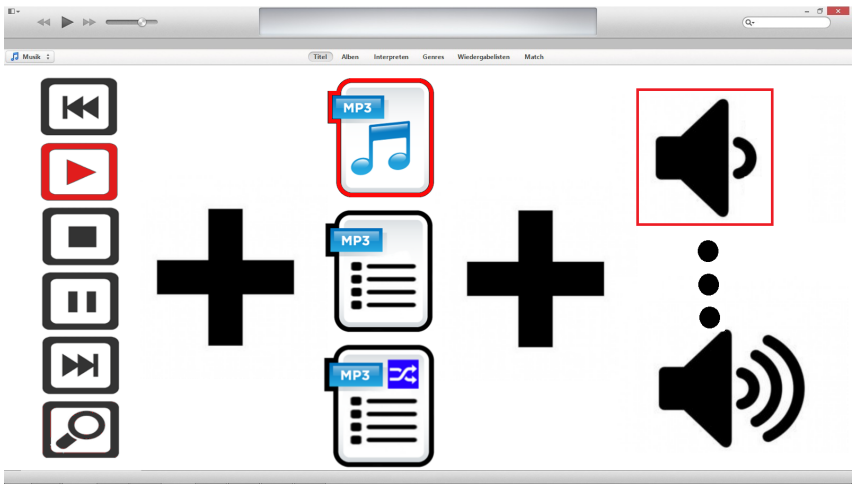


Abbildung 16: Problematisches Aufgabenblatt 10

Ziel des Aufgabenblattes ist es gewesen, einen ruhigen Song abzuspielen, da der rechte Bereich dieser AGSt die Musik beschreibt nach Stimmung, Geschwindigkeit etc. Doch viele brachten dieses Logo, im Zusammenhang mit dem Thema Musik, mit der Regelung der Lautstärke in Verbindung. Diese Darstellung war somit doppeldeutig und da sie bei vielen technischen Geräten als Abbildung für die Lautstärkeregelung genutzt wird, oft falsch interpretiert. Auch wenn die Kandidaten, die Problemen mit der Darstellung hatten, sie für ungeeignet hielten, fiel ihnen auch keine bessere und eindeutigere Darstellung ein, die für alle verständlicher wäre.

Eine Aufgabenstellung, die vorher mit allen Teilnehmern besprochen wurde, aber trotz

allem selten richtig verstanden wurde, war das Aufgabenblatt 7. Zum einen enthielt es das besprochene Icon, das wichtig war zur Bewältigung von 2 Aufgabenblättern und zum anderen wurde versucht mit einem Bild das Wort Genre darzustellen. Ein einzelnes Bild hätte hier, wie in den Erstversuchen, sicher nie zum richtigen Ergebnis geführt, weshalb versucht wurde, es mit Symbolen für Interpret und Album, den üblichen Tags eines Titels, in Verbindung zu bringen.

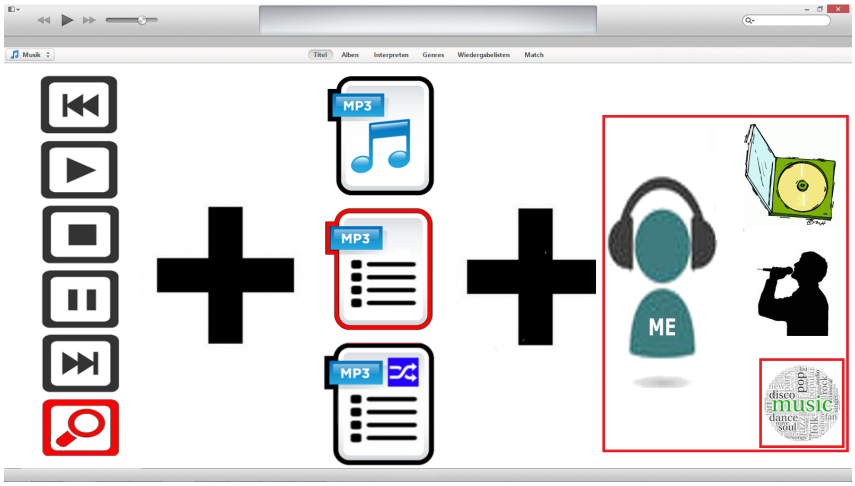
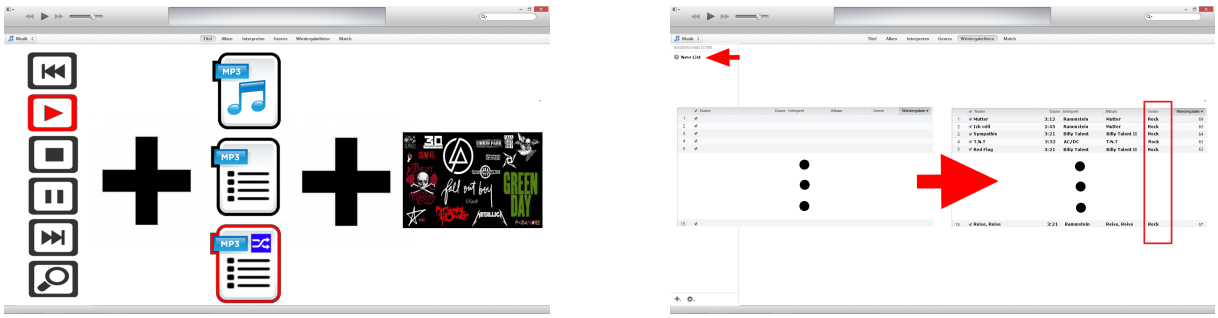


Abbildung 17: Problematisches Aufgabenblatt 7

Doch trotz Vorbesprechung kamen viele unter *'Druck'* nicht mehr zum richtigen Ergebnis. Nur 10% bewältigten es noch wie besprochen, der Rest interpretiert es vollkommen neu. Auf Grund dieses Ergebnisses muss man feststellen, dass manches in dieser Form der Aufgabenstellung einfach nicht bildlich Darstellbar ist. Die Konsequenz daraus ist, dass die Daten zu diesem Aufgabenblatt nicht zur Entwicklung der Grammatik berücksichtigt werden konnten und nur für die Variation der Vokabeln im Lexikon dienten.

Auf die Frage *'Welche Variante verständlicher war und warum?'* kam kein sonderlich überraschendes Ergebnis bei raus. Es haben sich 10 von 14 Probanden dafür entschieden, das die erste Variante (siehe Kapitel 5.2.3) *verständlicher* ist. Dabei ist aber zu erwähnen, das beide Aufgabenstellungen sehr gut angenommen wurden. Warum das Ergebnis nicht überraschend kommt, hat den Grund, dass hier die Probanden verständlicher mit einfacher gleichgesetzt haben und sich daher für die Aufgabenform entschieden haben, die einfacher zu lesen ist. Mit dem syntagmatische Zusammenhang bei der ersten Art der Aufgabenstellung, wird eine leichte Syntax vorgegeben. Hinzu kommt, dass die zweite Variante von den 14 Aufgaben nur 3 Aufgabenstellungen ausgemachte und durch das häufige Auftreten der Anderen, sie am Ende schneller zu bearbeiten ging. Doch um eine objektive Bewertung als Operateur gewinnen zu können, wurde ein Aufgabenblatt in beiden Varianten entwickelt, welches das selbe Ergebnis erzielen sollte.



(a) Aufgabenblatt 3 (b) Aufgabenblatt 9

Abbildung 18: Gleiche AGSt in beiden Varianten

In beiden Varianten sollten die Probanden zu dem Ergebnis kommen, eine Liste zu erstellen bzw. abzuspielen die in jedem Fall ein bestimmtes Genre enthält oder ausgewählte Bands. In Abbildung 17 (a) wurde zusätzlich verlangt, dass die Liste eine zufällige Wiedergabe ist. Bei dieser Variante ist es auch so, dass dem Kandidaten eine größere Freiheit gelassen wurde. Sie konnten dazwischen entscheiden eine Liste nach dem Genre oder der gezeigten Bands zu wählen, was auch 70% getan haben, wobei es zwischen Metal und Rock beim Genre variierte, auf Grund dessen, dass es nicht ihre Musikrichtung war oder eben einzelne Bands angeben, die in die Liste gepackt werden sollen. Jedoch schien die größere Freiheit mehr Fehler zu provozieren. Während bei Aufgabenblatt 9 100% die Aufgabe richtig lösten, wobei nur einer eine grammatikalisch unkorrekte Anweisung formulierte, es bei der anderen Variante nur 80% waren. Obwohl die Teilnehmer es in den anderen Aufgabenblättern richtig umgesetzt hatten, wählten sie hier einen Titel statt der Liste und weil die Band Green Day auf dem Bild zu auffällig war und direkt in den Fokus fiel, kamen Eingaben dabei raus, wie *'Computer, spiel einen zufälligen Titel von Green Day.'* Natürlich stand es dem Probanden frei, einfach nur Green Day zu wählen, aber was das Beispiel zeigen soll, ist, dass bei der AGSt in Abbildung 18b, die Kandidaten zu jeder Zeit wussten, dass sie mit einer Liste arbeiten, was in der anderen Variante scheinbar nicht der Fall war.

Bei der Überlegung, welche Aufgaben beim System zusätzlich via Sprache gesteuert werden können, wurden die Probanden besonders kreativ. Es ist erstaunlich, dass keine Antwort doppelt war. Doch einige fanden auch, das zusätzliche Sprachbefehle nicht nötig wären. Dabei reichten die Ideen vom direkten Bewerten der gerade gehörten Songs, über das einbinden zusätzlicher Software (brennen, download etc.), bis hin zum selbstständigen erweitern des Systems. Da wurde speziell daran gedacht, den Spracherkenner selbstständig trainieren zu können und auch zusätzliche Tags setzen zu können, die vom System gleichzeitig erkannt werden. Eine vollständige Übersicht aller Erweiterung befindet sich im Anhang E.

6.3 Datenauswertung

Aus den aufgezeichneten Sprachaufnahmen, soll nun eine erste Grammatik entwickelt werden, sodass es dem Spracherkenner möglich sein sollte, alle Eingaben, die gemacht wurden, zu verstehen. Hierzu wurden die Aufnahmen von Hand geschnitten und beschriftet.

Die Ergebnisse waren zum großen Teil positiv, denn die Versuchskandidaten stellten ihre Anweisungen in vollständigen Sätzen, wie vorab vom Operateur gewünscht.

Bei der ersten Betrachtung der gewonnen Daten viel auf, dass sich durch die deutsche Grammatik, im Aufbau des Satzes nicht so viele Variationen bilden lassen. Doch gab es auch Ausnahmen, denn durch die nicht reale Umgebung und den *'Druck'* auf den Probanden, eine vorgegebene Anweisung zu formulieren, gab es auch unnatürliche Eingaben, die für unsere Grammatik nicht verwendet werden. Außerdem wurden alle Eingaben erstmal nicht berücksichtigt, die nicht zur richtigen Erfüllung der Aufgabenstellung führten. Was sich auch bei der ersten Betrachtung und schon durch das System bestimmen lässt, ist jede Anweisung abhängig vom Verb, welches bestimmt, was wir machen wollen. Mit dem Verb als zentralen Punkt unserer Grammatik, wurde der Satz drum herum in seine einzelnen Elemente zergliedert. Das Vorgehen zur Bestimmung einer Grammatik bezeichnet man als Dependenzgrammatik.

Dependenzgrammtik:

Für die Dependenzgrammatik sind die Dependenzen oder Abhängigkeiten zwischen den einzelnen Elementen des Satzes bestimmend, sie erfasst die Syntax der meisten natürlichen Sprachen in Form von Abhängigkeitsverhältnissen unter den Elementen eines Satzes, den Wörtern. Zwischen den Wörtern bestehen Dependenzbeziehungen, diese sind dabei mehrfach miteinander verbunden. Sie stehen also nicht nebeneinander, wie in einem Wörterbuch [Neu13]. Wenn man Sätze betrachtet, gibt es dabei ein sogenanntes regierendes Wort, dass wie schon erwähnt das Verb ist, von dem die anderen Elemente in abhängig sind. Das Verb regiert gewissermaßen den Satz, indem es bestimmte Leerstellen für die beteiligten Nomen bestimmt. Dadurch zeichnen sich Verben durch eine bestimmte Wertigkeit oder Valenz aus, die die Anzahl der möglichen *Aktanten* oder Mitspieler bestimmt. Die *'Aktanten'* sind Substanzen, also Lebewesen oder Dinge, die aktiv oder passiv an durch das Verb beschriebenen Aktionen beteiligt sind (Subjekt, Objekt). Wörter bilden das Inventar zur Ausgestaltung von Sätzen. Man unterscheidet der Dependenz- oder Valenzgrammatik zwei Arten von Wörtern:

1. *leer-Wörter*: Das sind Wörter, die für die Einhaltung der sprachlichen Grammatik zwar wichtig sind, aber keine *'lexikalische'* Bedeutung haben, das heißt nur als Füllwörter fungieren.

2. *voll-Wörter*: Sie drücken eine Vorstellung aus, sie haben deshalb eine syntaktische als auch semantische Funktion. Dabei unterscheidet man zusätzlich unter vier unterschiedlichen Wortarten

    a) Verben = konkrete Prozesse

b) Substantive charakterisieren und bezeichnen die Objekte

c) Adjektive charakterisieren die Objekte

d) Adverbien bezeichnen hingegen die Prozesse in abstrakter Form [Hoc04]

Das bedeutet das in unserer Grammatik das Verb das *regierende* Wort ist, was erst einmal beschreibt, welche Funktion wir ausgeführt haben wollen. In all unseren Daten stand dieses im Hauptsatz immer am Anfang oder am Ende des Satzes und sieht somit sieht das in der Grammatik folgender Maßen aus:

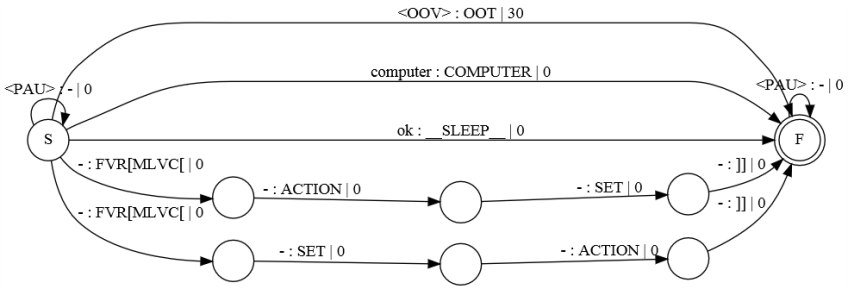


Abbildung 19: Auszug Grammatik

Das Verb beschreibt die Aktion (ACTION), die ausgeführt werden soll und der restliche Satz (SET) wird in den unteren Ebenen der der Grammatik weiter aufgeschlüsselt. Dabei werden nur *'Voll-Wörter'* berücksichtigt, also nur die Wörter die wichtige Informationen für das System enthalten.

Der eine Teil der Variation liegt in der Grammatik, doch der weit wichtigere Anteil ist die Variation durch das Vokabular einer Sprache. Dies lässt sich an einem der einfacheren Aufgaben betrachten. Im Anhang D das Aufgabenblatt 6 wurde von allen 14 Probanden erfolgreich bewältigt. Da einige Teilnehmer diese selbe Anweisung verwendet haben, sind hier keine Dopplungen aufgeführt und somit keine 14 Varianten. Das Aufgabenblatt sollte den Tester dazu bringen eine der folgenden Anweisungen abzugeben:

1. Computer spiele ein trauriges Lied ab.
2. Computer spiele einen traurigen Song.
3. Computer spiele einen Titel ab, der traurig ist.
4. Computer bitte spiele mir einen traurigen Song.
5. Computer gebe mir ein trauriges Lied wieder.
6. Computer ich möchte gerne ein einzelnes Lied, was relativ traurig ist, hören.
7. Computer spiel mir ein trauriges Lied.
8. Computer spiele mir bitte einen traurigen Song ab.

Alle farbig markierten Wörter sind jene mit einer Information, also die in einer Dependenzgrammatik bezeichneten *Voll-Wörter*. Außer dem Aktivierungswort *'Computer'*, sind somit alle nicht farbigen Wörter *Leer-Wörter* und haben für die Syntax und den Aufbau unserer Grammatik keinen Wert. Bei solch einfachen Beispielen ist sowohl die grammatikalische, als auch die Variation an Vokabeln eher gering. So haben wir als Objekt(OBJECT) Titel, Song oder Lied, als Verb(ACTION) spiel/e, hören und gebe/wieder oder sogar keine Unterschiede in den verwendeten Adjektiv (ADJ). Wenn man sich aber aufwendigere Anweisungen betrachtet erhöht sich die Variation, besonders im Vokabular. Dies können wir nur an einem kurzen Beispiel betrachten an Hand von Aufgabenblatt 2, in welchem der zuletzt gehörte Titel an den Einfang einer Liste einsortiert werden sollte.

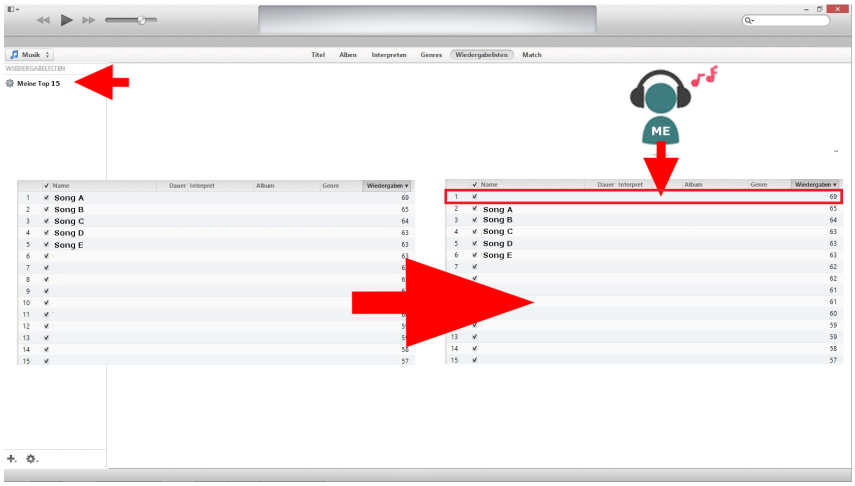


Abbildung 20: Aufgabenblatt 9

1. Computer füge den gerade gehörten Titel in meine Top fünfzehn an Platz eins ein.
2. Computer packe den eben gespielten Song in die Liste meine Top fünfzehn an Stelle eins.
3. Computer füge den gerade gehörten Song am Anfang der Liste meiner Top fünfzehn ab.
4. Computer füge aktuellen Titel in die Wiedergabeliste meine Top fünfzehn an erster Position ein.

Hier sieht man, dass allein für die Formulierung der Position, mehrere Möglichkeiten vorhanden sind. Wobei die Grammatik sich in allen Varianten ähnelt.

Die Betrachtung der restlichen Aufgabenstellungen hat gezeigt, dass der Aufbau aller Eingaben sich ähnelt, was mit daran liegen kann, dass durch die eine Variante der AGSt zu sehr eine Syntax vorgegeben wird und, da diese Formulierung funktionierte, insgesamt verwendet wurde. Damit wurde das System nicht voll ausgenutzt, weil somit die Eingaben *relativ* einfach formuliert wurden. Die gewonnene erste Grammatik, die aus diesen Eingaben erstellt wurde, kann man im Anhang H anschauen. Sie ist noch keine fertige Version, muss in jedem Fall verfeinert werden und wird sich, mit der Weiterentwicklung des Systems, sicher noch wachsen und verändern.

7 Fazit und Ausblick

Bei der Entwicklung von Programmen und Systemen, die direkt auf Benutzereingaben reagieren müssen, sind *Wizard of Oz*-Versuche eine gute Möglichkeit um verschiedene Design-Möglichkeiten zu probieren, ohne sie zuerst programmieren zu müssen. Ein Tester kann innerhalb der vorgegebenen Parameter und der bereits gestalteten Programmoberfläche ganz nach Belieben mit dem System oder dem Programm arbeiten und seine Eingaben tätigen. Durch eine solche user-orientierte Entwicklung, bei der nicht die Entscheidungen und Wünsche von einzelnen Designern oder Entwicklern im Vordergrund stehen, sondern die Nutzungsgewohnheiten von späteren System- und Programmbenutzern. Des weiteren lassen sich natürlich auch bei abschließenden Befragungen und Gesprächen Schwächen der genutzten Oberfläche herausfinden und verbessern.

Insgesamt sehe ich das 'Wizard of Oz'-Konzept zum Testen von Programmen sehr positiv, jedoch müssen solche Tests auch gut vorbereitet und betreut werden und es sollte mehr Zeit zur Verfügung stehen, solche Experimente vorzubereiten. In dieser Bachelorarbeit wurde ein Großteil der Zeit dafür investiert, die Aufgabenstellung zu entwickeln, da diese Variante des Experiments bisher so nicht existierte. Auch meine Internetrecherche ergab keine ähnlichen Versuchsdurchführungen, weshalb alles von Grund auf selbst entwickelt wurde. In diesem ersten von mir durchgeführten Experiment, musste ich schnell feststellen, dass die Methode auch ihre Grenzen hat, wenn es um die bildliche Darstellung geht. Auch wenn beide der Aufgabenblätter gut funktioniert haben, außer einer Ausnahme, ist schon bei der Entwicklung aufgefallen, dass sich nicht alles bildlich darstellen lässt. An manchen Stellen sollte man, meiner Meinung nach, lieber auf Text zurück greifen, um eine eindeutig Darstellung zu schaffen. Ich finde auch, dass die Aufgabenstellung nach dem ABC-Schema, die bessere Variante war. Zum einen gab es keine Symbolik durch Bilder, die man fehlinterpretieren könnte und zum anderen wird bei dieser Form der Aufgabenstellung dem Testkandidaten keine Syntax vorgegeben oder wie ich es sogar empfinde, aufgezwungen. Denn zur Erreichung einer hohen Variation der Sprache, sollten dem Probanden mehr Freiheiten gelassen werden in der Formulierung seiner Systemeingaben. Außerdem muss ich anmerken, dass auch wenn die Tester die erste Variante zum Großteil als leichter verständlich beschrieben haben, bei der Variante des ABC-Schemas **keiner** ein Problem gehabt hat, bei der Bewältigung der Aufgabenstellung. Lediglich die Vorüberlegung, wie man die Systemeingabe formuliert, dauerte etwas länger, weil es nicht so leicht zu lesen war. Bei der Durchführung des Experimentes fiel auf, das die Simulation eines existierenden Systems sehr gut funktionierte. Das mache ich nicht nur daran fest, das ich als Wizard nur einmal erkannt wurde, sondern auch an den vorher ausgewerteten Daten. Viele der Probanden nutzen eben nur relativ kurze Formulierung, um die Wünsche an das System zu stellen. Vorab wurde ihnen von mir aber verdeutlicht, dass sie mit dem System umgehen können, als wäre es ihr persönlicher Butler, der alles verstehen sollte und ihre Musikwünsche erfüllt. Doch wirklich ausgereizt hatte dies sprachlich nur ein Proband. Ich denke diese kurzen Systemeingaben liegen mit daran, dass die Probanden, die alle technisch versiert waren, durch bekannte Player und sprach-gesteuerte Systeme schon vorgeprägt waren. Auch wenn die Mensch-Maschine-Kommunikation ein Grundwissen zum technischen Gerät voraussetzt, wären weniger technisch versierte Probanden für

die weitere Entwicklung empfehlenswert. Sollte sich nichts an der technischen Ausstattung im Sprachlabor verändern, ist der Aufbau so ganz gut. Abstriche beim Aufbau mussten nur beim Anschluss von Mikrofonen machen, weil es nicht möglich war, zusätzlich mit einem Headset aufzunehmen. Die größte Problematik bei der Durchführung solcher Experimente, sehe ich nicht bei der Mensch-Maschine-Kommunikation, sondern in dem nicht realen Umfeld, wo die Experimente stattfinden. Dadurch das die Probanden nicht mit ihrer eigenen Musikbibliothek arbeiteten und nicht ihre eigenen Wünsche, sondern die durch die Aufgabenstellung vorgegebenen, äußern mussten, sind die Systemeingaben nicht ganz realitätsnah. Die gewonnenen Daten sind aber in jedem Fall hilfreich zur Entwicklung der Sprachsteuerung, doch sind mit Sicherheit noch mehr Durchläufe an Experimenten nötig, um ein gut funktionierendes System zu entwickeln. Für die zukünftige Entwicklungen in dieser Richtung sollte also berücksichtigt werden, das die Vorgaben an Aufgabenstellungen, die mögliche Eingabe des Testers nicht zu sehr einschränkt, um eine realitätsnähere Nutzung des Systems zu simulieren. Hierfür könnte es von Vorteil sein mehr als nur einen Wizard einzusetzen, wenn die Antwortmöglichkeiten eine Person zu sehr fordern sollten. Dies findet auch Anwendung bei Wizard-of-Oz Experimenten mit vielen Funktionen.

Zusammenfassend würde ich also sagen, dass dies ein guter Erstversuch war, doch ist eine allgemeine *'Schablone'* für künftige Experimente noch nicht wirklich geschaffen. Allerdings, finde ich, ist das ABC-Schema, wie hier angewendet, sicher auch für andere Wizard-of-Oz Versuche ein gutes anwendbares Schema. Aber für die Entwicklung dieser Sprachsteuerung ist eine gute Basis vorhanden. Es existiert eine erste Grammatik, die erweitert werden kann und Schemen die weiterentwickelt werden können, da beide gut funktionierten. Beim Aufbau ist mit dem vorhandenen Mitteln die bestmögliche Zusammenstellung von Hardware und Software getroffen worden.

8 Quellenverzeichnis

[20008] *Semiotisches Dreieck*. `http://upload.wikimedia.org/wikipedia/commons/f/ff/Semiotischesdreieck.jpg`. Version: 2008

[Dum12] Dumbill, Edd: *What is big data? An introduction to the big data landscape.* `http://radar.oreilly.com/2012/01/what-is-big-data.html`. Version: Januar 2012

[Fra91] Fraser, G. N. N. M. & Gilbert G. N. M. & Gilbert: Simulation speech systems. In: *Computer Speech and Language* (1991)

[Gre04] Green, Hüttenrauch H. & Eklundh K. S. A.: Applying the Wizard-of-Oz Framework to Cooperative Service Discovery and Configuration. In: *In Proceedings of the 2004* (2004)

[Hoc04] Hochhaus, Stephan: *Dependenz und Valenzgrammatik*. `https://www.yauh.de/files/klausur_dg.pdf`. Version: 2004

[Hru10] Hruby, Florian: *Kartosemiotik und geographisch-räumliche Abbildungen*. `http://www.inst.at/trans/17Nr/2-10/2-10_hruby17.htm`. Version: März 2010

[Kar12] Karnagel, Kathleen: *Semantische Modellirung von System-Benutzerinteraktionen am Beispiel einer Home-Entertaiment Steuerung*, BTU Cottbus, Diplomarbeit, 2012

[Kel84] Kelley, J. F.: An iterative design methodology for user-friendly natural language office information applications. In: *ACM Transactions on Information Systems* (1984), S. 26-41. `http://dl.acm.org/citation.cfm?id=357420`

[Kjo09] Kjørup, Søren: *Simiotik*. W. Fink Verlag, 2009

[KMB08] Klaus Michael Bogdal, Georg M. Kai Kauffmann K. Kai Kauffmann: *Ba-Studium Germanistik: ein Lehrbuch*. Bd. 344 Seiten. Rowohlt-Taschenbuch-Verlag, 2008

[Koc10] Kocsány, Piroska: *Grundkurs Linguistik: ein Arbeitsbuch für Anfänger*. Fink Verlag, 2010

[Lew94] Lewandowski, Theodor: *Linguistisches Wörterbuch*. Quelle & Meyer, 1994

[Mic14] Microsoft: *Windows Media Player*. `http://windows.microsoft.com/de-de/windows/windows-media-player`. Version: 2014

[Neu13] Neumann, Günther: *Einführung in die Dependenzgrammatik*. `http://www.coli.uni-saarland.de/courses/grammatikformalismen/2013/slides/dg.pdf`. Version: 2013

[PRC94] Philip R. Cohen, Sharon R. O.: *The role of voice input for human-*

machine communication. http://www.sri.com/sites/default/files/uploads/publications/pdf/1400.pdf. Version: January 1994

[Rie12] Riek, L. D.: Wizard of Oz Studies in HRI: A systematic review and new reporting guidelines. In: *Journal of Human-Robot Interaction* (2012), S. 119-136. http://www.humanrobotinteraction.org/journal/index.php/HRI/article/viewFile/9/26

[Rud06] Rudlof, Christiane: *Handbuch Software Ergonomie - Usability Engineering.* http://www.ukpt.de/pages/dateien/software-ergonomie.pdf. Version: 2006

[Ski91] Skinner, B. F.: *Verbal Behavior.* Taschenbuch, 1991

[Ste] Stepken, Guido: *Linux Solution Guide.* http://www.little-idiot.de/linuxsolutionguide/guigestaltung.htm

[Ste09] Steinfeld, Jenkins S.: *The Oz of Wizard: Simulating the human for interaction research.* http://scholar.google.com/citations?view_op=view_citation&hl=de&user=CQ39OowAAAAJ&citation_for_view=CQ39OowAAAAJ:5nxA0vEk-isC. Version: 2009

[Ulr72] Ulrich, Winfried: *Linguistische Grundbegriffe.* Taschenbuch, 1972 http://www.noirscript.de/dramaturgie/relatio.htm

[wir13] wirzberger maria: *Wizard of Oz Szenarien.* 2013

Anhang

A Fragebogen Experiment

Brandenburgische Universität
Cottbus-Senftenberg
Fakultät 3
Lehrstuhl Kommunikationstechnik

WoO-Fragebogen

Sprachsteuerung einer Musikbibliothek

Wie finden Sie hat das System gearbeitet?

Hätten Sie nach Ihren Eingaben, eine andere Ausgabe durch das System erwartet?

Waren die Aufgabenblätter alle verständlich? Wenn nicht, wo gab es Probleme?

Welche Variante war verständlicher und warum?

Fehlten Aufgaben, die das System übernehmen können sollte, um Ihre Musikbibliothek vollständig zu steuern?

B Handout Experiment

Brandenburgische Universität
Cottbus-Senftenberg
Fakultät 3
Lehrstuhl Kommunikationstechnik

Brandenburgische Technische Universität Cottbus - Senftenberg

Lehrstuhl Kommunikationstechnik

Handout

Sprachsteuerung einer Musikbibliothek

Wizard-of-Oz Experiment

Im Sprachlabor des Kommunikationstechnik Lehrstuhls ist als Anwendungsbeispiel, eine Spracherkennung zur Steuerung eines Mediaplayers entwickelt worden. Hierzu wurde eine Grammatik erstellt, welche in diesem Experiment mit einer Auswahl an Probanden getestet werden soll.

Versuchsablauf:

Das Ziel des Experimentes ist es, dass Sie mittels Ihrer natürlichen Sprache einen üblichen Mediaplayer (Windows Media Player) steuern. Versuchen Sie keine Befehle zu geben, sondern möglichst flüssig zu sprechen, als wäre da eine andere Person, welche Ihre Musikwünsche erfüllen soll. Hierzu betrachten Sie die Aufgabenblätter gut, überlegen Sie erst, wie Sie eine solche Aufforderung stellen würden und sprechen Sie dann laut und deutlich Ihre Wünsche aus. Damit das System aber weiß, dass es angesprochen wird, geben Sie vor jeder Aufforderung das Hinweiswort ***„COMPUTER“*** an. Es ist nötig möglichst wenig Füllungen, wie „ähm“ oder „ähh“, zu verwenden. Vorab werden wir 2 Aufgabenblätter testweise behandeln, um zu schauen ob die Vorgaben verständlich sind. Diese werden auch aufgenommen, doch danach beginnt das eigentliche Experiment und Sie bewältigen die restlichen Aufgabenblätter selbstständig. Sollten dem System Informationen fehlen, erscheint ein Pop-Up Fenster mit Hinweisen, was benötigt wird. In dem Fall, dass ihre Eingabe richtig war und erkannt wurde, spielt kurz die Musik an, wird dann aber vom System pausiert, um den Testlauf nicht unnötig hinzuziehen.

Versuchsaufbau:

In dem Experiment werden Sie vor dem Bildschirm sein, mit einem Headset auf dem Kopf und einem Mikrofon vor sich, um Ihre Eingaben per Sprache zu tätigen und aufzuzeichnen. Sie erhalten eine Auswahl von 14 Aufgabenblättern, die die geforderten Wünsche an das System darstellen. Bei den Aufgabenblättern handelt es sich um *ausschließlich bildliche* Darstellungen, welche zu interpretieren sind.

Es sind 2 unterschiedliche Varianten an Aufgabenbildern zu bearbeiten, wobei bei beiden, das mit **ROT** gekennzeichnet ist, was verlangt wird:

1. In der ersten Variante ist aus den drei syntagmatisch verbundenen Auswahlbereichen, der vollständige Befehl ans System zu richten. In jedem der drei Bereiche finden sie senkrecht eine paradigmatische Auswahl an Symbolen, wo das **ROT** markierte, das jeweils verlangte ist.

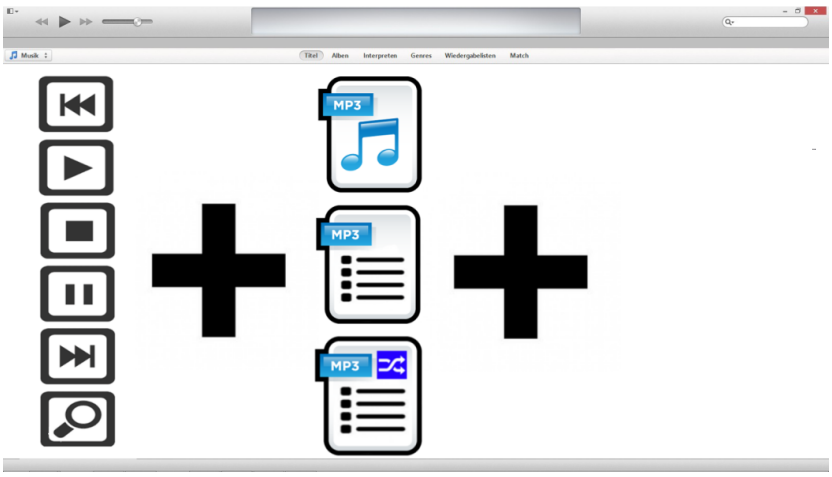


Abbildung 1

In *Abbildung 1* ist die Vorlage gegeben, die bei jedem dieser Blätter gleich sein wird. Im letzten Bereich wird dann immer ein Attribut dargestellt, wie das gewünschte Objekt sein soll.

2. Die andere Variante wird eine Vorher/Nachher Darstellung sein. Die linke Seite ist der derzeitige Zustand und die rechte, der neu gewünschte Zustand.

EINVERSTÄNDNISERKLÄRUNG

Hiermit erklärt sich der Proband dazu bereit, dass alle seine Spracheingaben aufgenommen und zu weiteren wissenschaftlichen Zwecken verwendet werden dürfen. Alle Daten werden anonym behandelt und ausschließlich für die Entwicklung des Spracherkenners am Lehrstuhl Kommunikationstechnik genutzt.

Bei Fragen zum Experiment und zur Nutzung der Daten, wenden Sie sich bitte direkt an:

Mike Stöhr Email: stoehmik@tu-cottbus.de Telefon: 01775673956

_____ _____

Datum / Unterschrift des Probanden

Datum / Unterschrift des Versuchsleiters

C Erstversuche AGSt

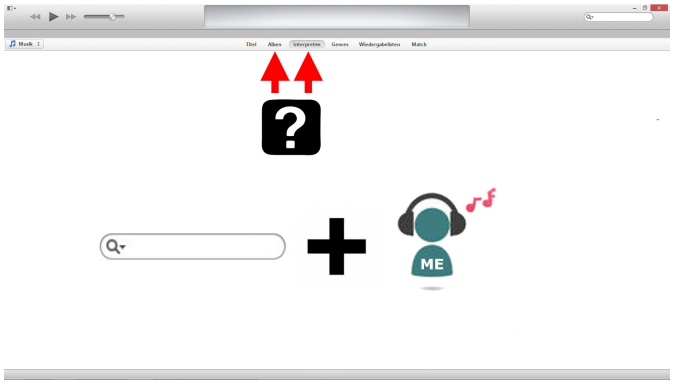


Abbildung 21: Suche alle Titel des grade gehörten Interpreten

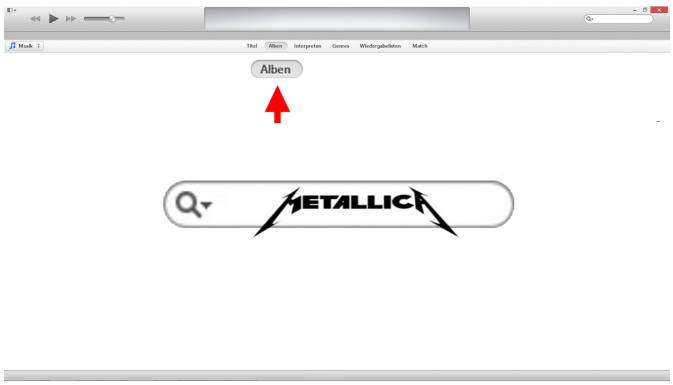


Abbildung 22: Suche alle Alben von Metallica

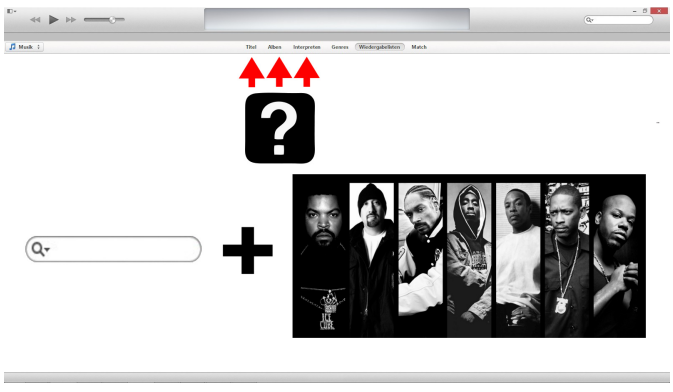


Abbildung 23: Suche alles zum Genre Rap/Hip Hop

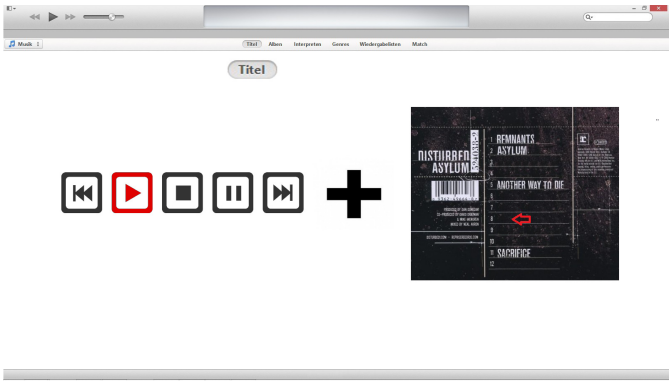


Abbildung 24: Spiele Titel 8 der Band Disturbed vom Album Asylum

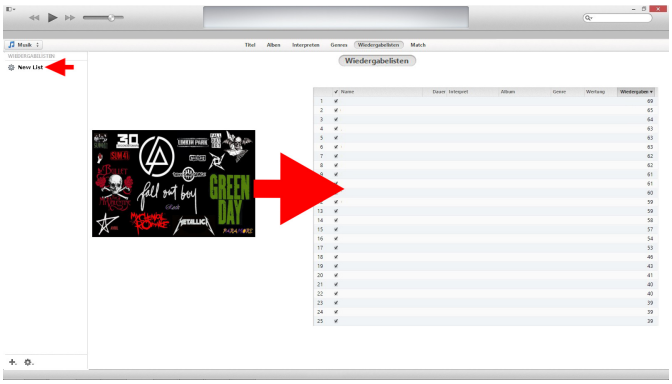


Abbildung 25: Neue Liste mit Roch / den zu sehenden Bands

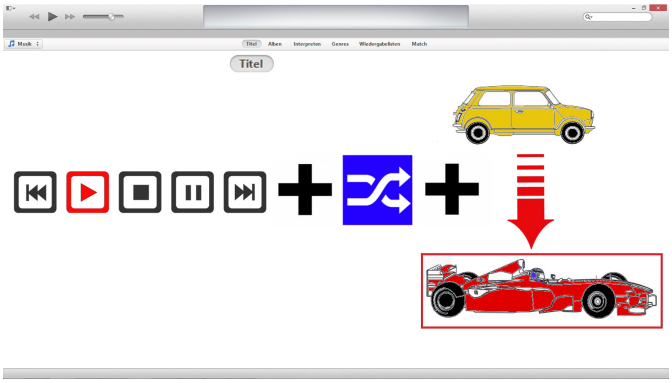


Abbildung 26: Abspielen zufälliger schneller Lieder

D Fertige AGSt

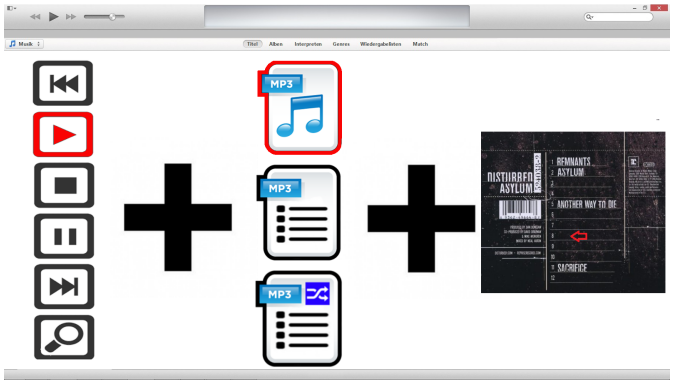


Abbildung 27: Aufgabenblatt 1

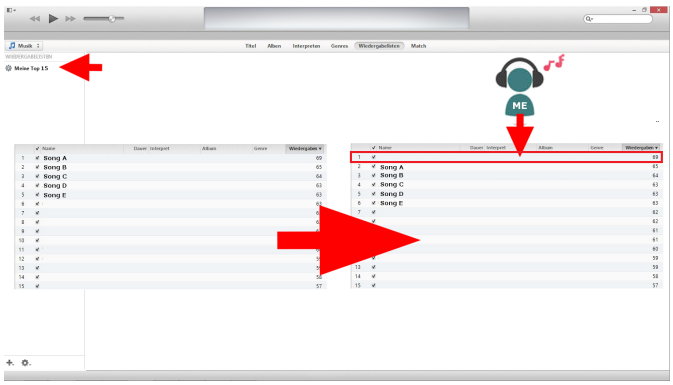


Abbildung 28: Aufgabenblatt 2

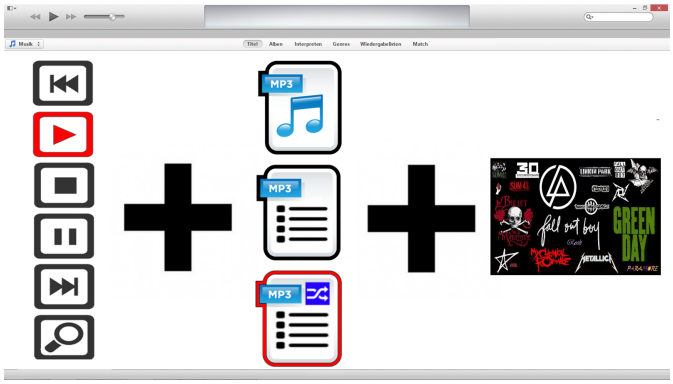


Abbildung 29: Aufgabenblatt 3

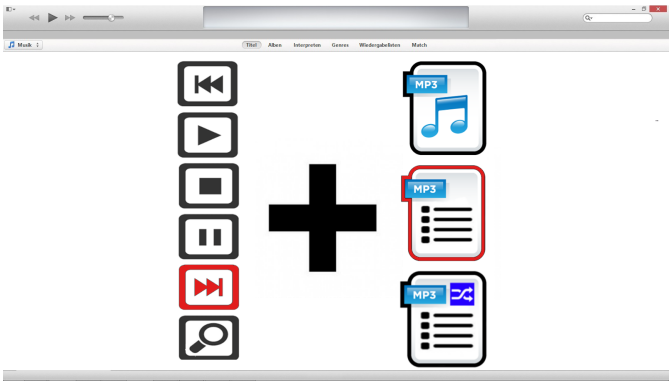


Abbildung 30: Aufgabenblatt 4

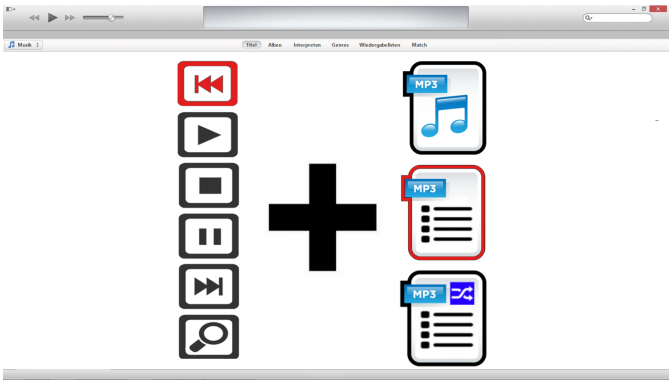


Abbildung 31: Aufgabenblatt 5

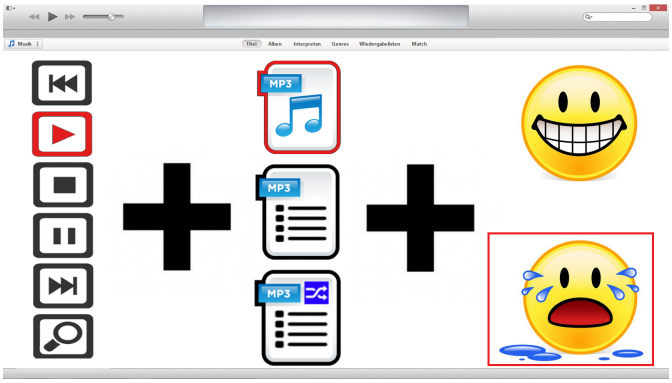


Abbildung 32: Aufgabenblatt 6

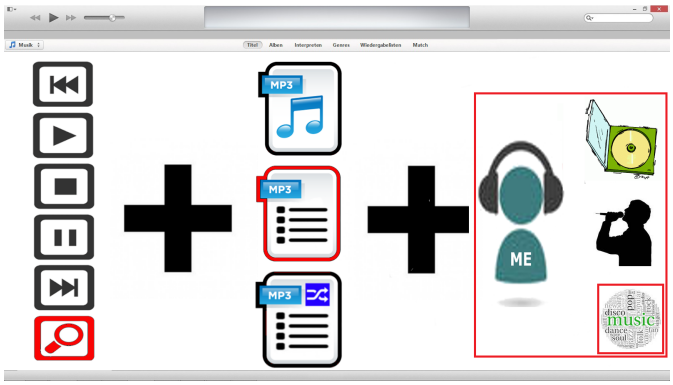


Abbildung 33: Aufgabenblatt 7

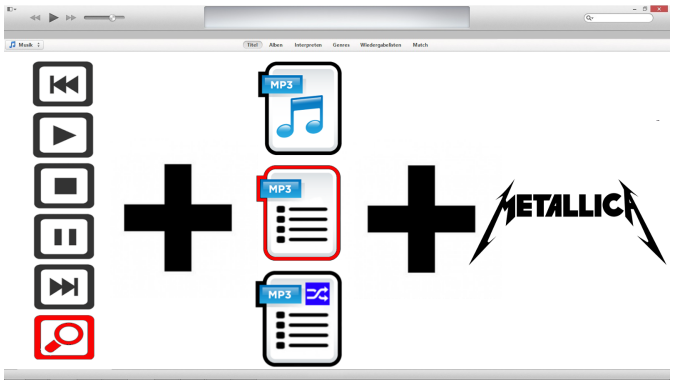


Abbildung 34: Aufgabenblatt 8

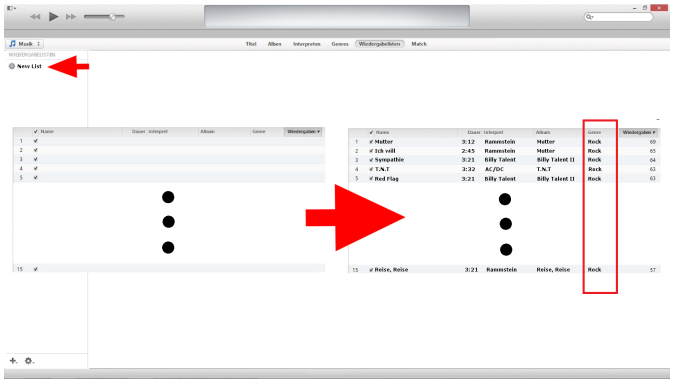


Abbildung 35: Aufgabenblatt 9

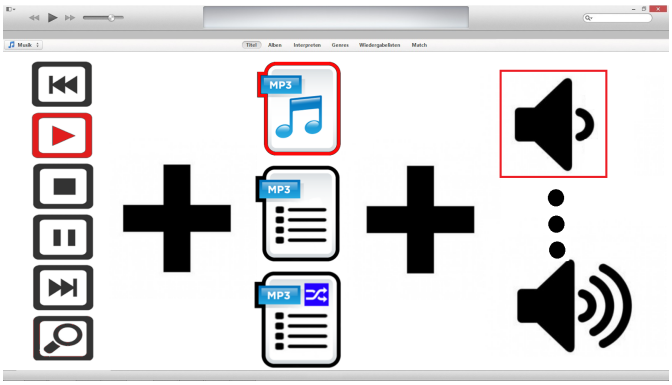


Abbildung 36: Aufgabenblatt 10

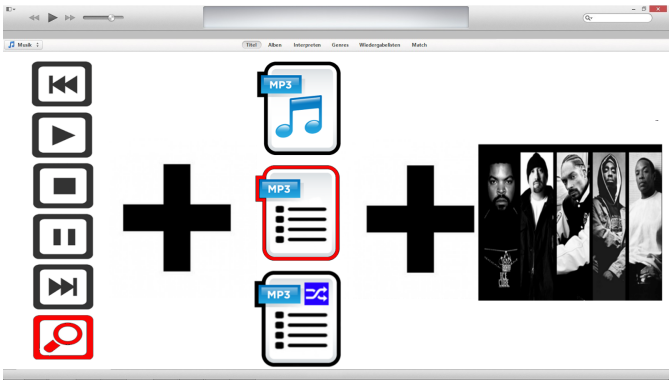


Abbildung 37: Aufgabenblatt 11

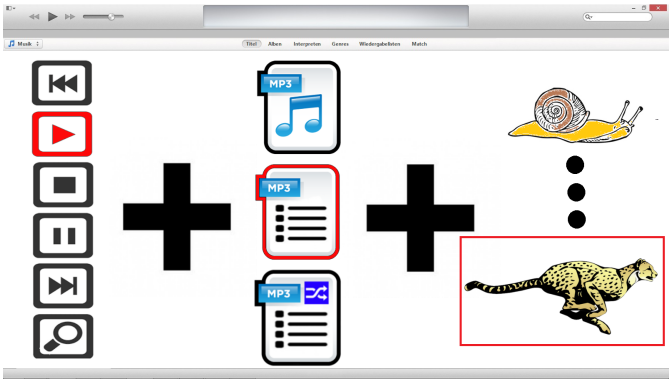


Abbildung 38: Aufgabenblatt 12

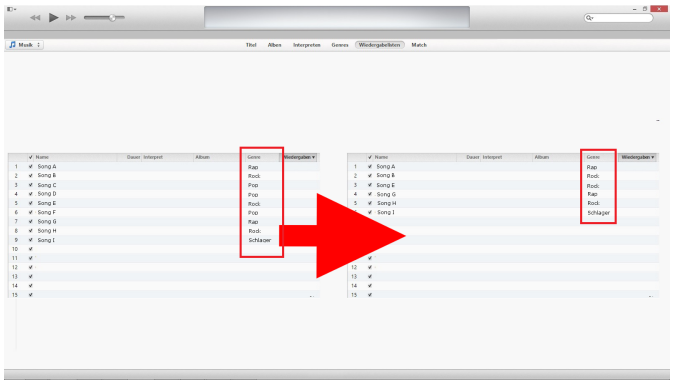


Abbildung 39: Aufgabenblatt 13

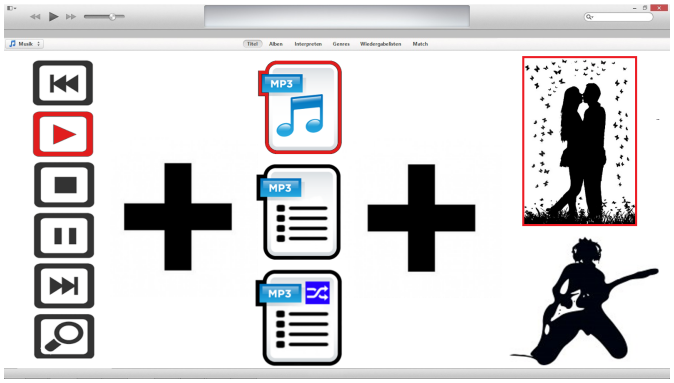


Abbildung 40: Aufgabenblatt 14

E Weitere sprachgesteuerte Aufgaben fürs System

1. Skits in der Bibliothek markieren, damit sie nicht mit abgespielt werden
2. Mehrere Wiedergabelisten untereinander shuffeln
3. Wiederholung von bestimmten Songs
4. Lyrics zu Songs suchen/anzeigen im Fenster
5. Brennoption, Listen direkt auf CD-Rom brennen
6. die Liste bearbeiten/umbenennen/ nach Attributen sortieren
7. Listen auf bestimmte Zeitlängen begrenzen
8. beim hören Titeln Tags zufügen
9. Tools anbinden zum bearbeiten von Songs (DJ Tools)
10. direktdownload von Songs, wenn einem ein Titel gefällt (Auswahl nach ähnlichen Titeln oder selber Interpret)
11. eigene Tags setzen, die das System auch erkennt
12. Pausieren per Sprachbefehl
13. während des Hörens bewerten

F Liste mit Schnelltastenauswahl

Brandenburgische Universität

Cottbus-Senftenberg

Fakultät 3

Lehrstuhl Kommunikationstechnik

WoO Tastenabfolge

1 Blatt: **ALT + A**

2 Blatt: **ALT + B**

3 Blatt: **ALT + C**

4 Blatt: **Strg + F**

5 Blatt: **Strg + B**

6 Blatt: **ALT + D**

7 Blatt: **ALT + E (DANN) ALT + L**

8 Blatt: **ALT + C**

9 Blatt: **ALT + G**

10 Blatt: **ALT + 1**

11 Blatt: **ALT + E DANACH ALT + H (RAP) ODER ALT + I (Hip Hop)**

12 Blatt: **ALT + J**

13 Blatt: **ALT + 7**

14 Blatt: **ALT + K**

WICHTIGE TASTENKOMBOS:

Strg + P → Track pausieren

Alt + F4 → Programm schließen

Win + Zahl → in Taskleistenprogrammen wechseln

Strg + E → Suchfeld WMP

Strg + 1 → Bibliotheksansicht WMP

ALT + S → Marker setzen Audition

KOMMENTARTASTEN:

ALT + 1: Das System hat keinen weiteren Eintrag gefunden!

ALT + 2: Das System benötigt mehr Informationen zum Interpreten!

ALT + 3: Das System benötigt mehr Informationen zum Album!

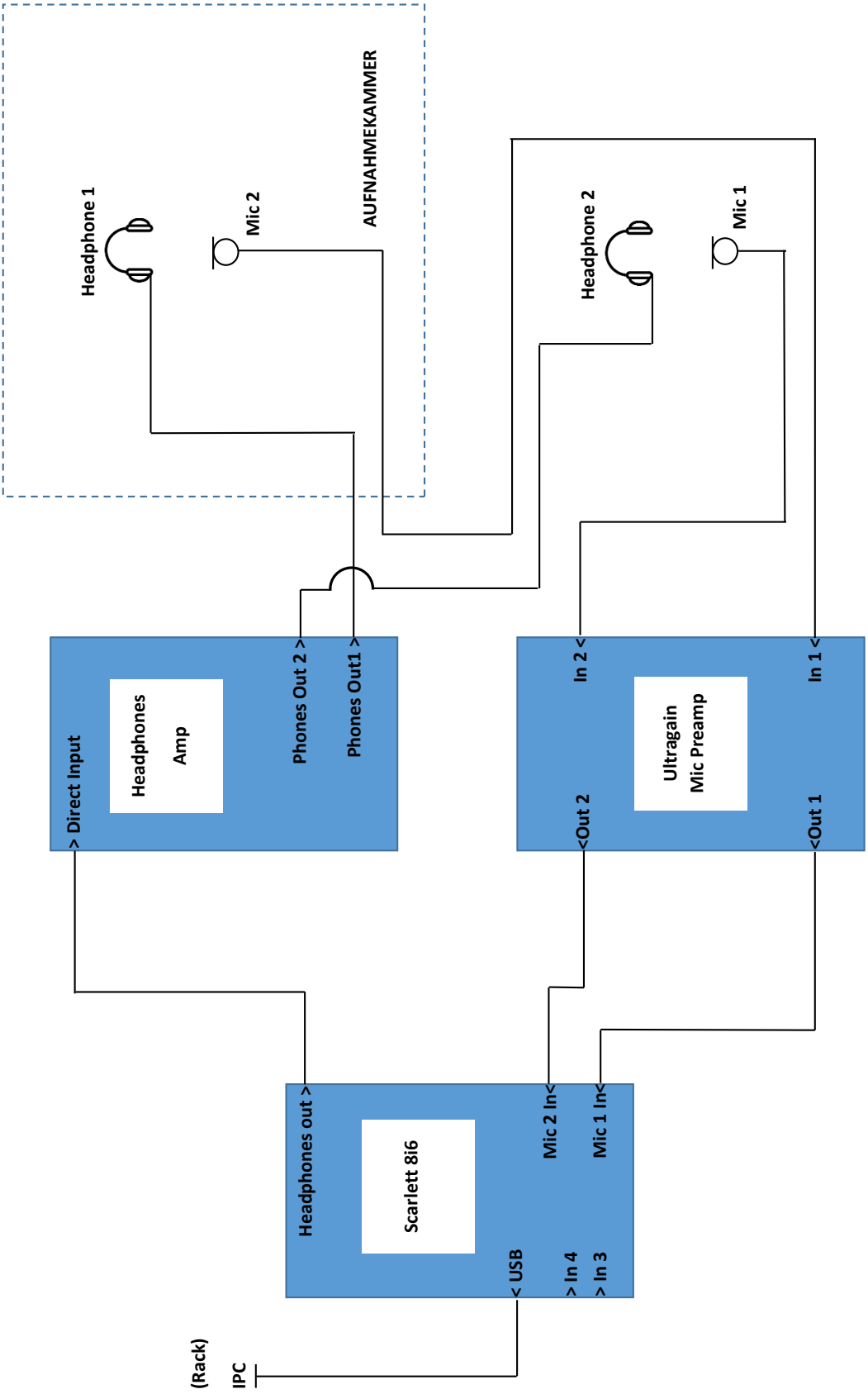
ALT + 4: Das System benötigt mehr Informationen zum Genre!

ALT + 5: Das System hat die Eingabe nicht verstanden, bitte wiederholen Sie Ihre Eingabe!

ALT + 6: Das System hat keinen Eintrag gefunden, bitte machen Sie eine andere Eingabe!

ALT + 7: Es wurden keine Elemente zum Löschen gefunden!

G Schaltplan Sprachlabor



H Grammatik

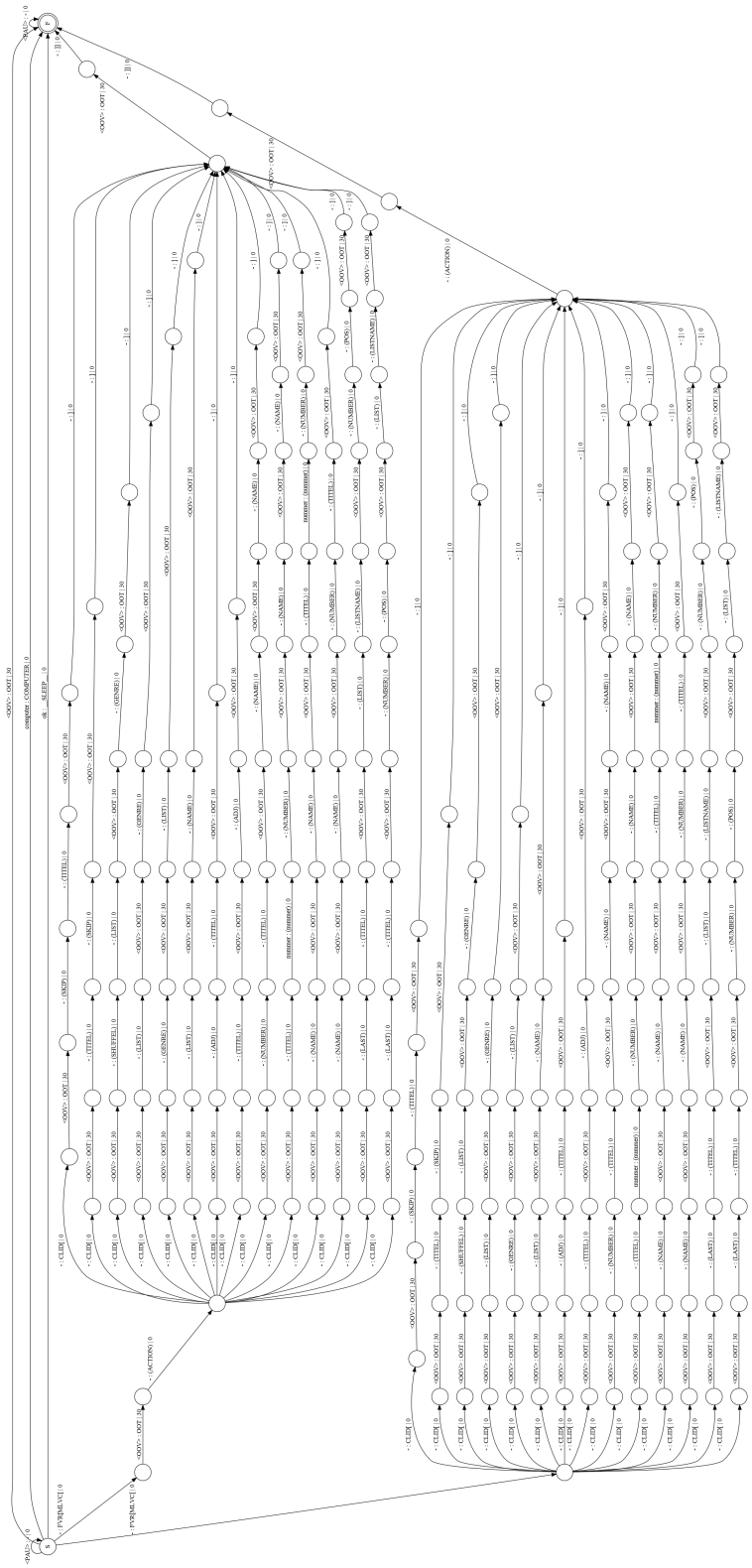


Abbildung 41: Grammatik