

Prüfer: Prof. Dr. rer. nat. Kurt Rothermel

Betreuer: Dipl.-Inform. Peter Kutschera
Dipl.-Inform. Frank Schiele

begonnen am: 01.08.1997

beendet am: 14.02.1998

CR-Klassifikation: H.2.8, J.4.2

Diplomarbeit Nr. 1584

World Wide Web basiertes Testen

Wolfgang Mahnke

Institut für Parallele und
Verteilte Höchstleistungsrechner
Universität Stuttgart
Breitwiesenstraße 20-22
D-70565 Stuttgart

Inhaltsverzeichnis

Vorbemerkung zur Gleichstellung der Geschlechter	vii
1 Einleitung	1
2 Grundlagen der pädagogischen Psychologie zur Durchführung und Bewertung von Tests	4
2.1 Was ist ein Test?	4
2.2 Welche Anforderungen werden an einen Test gestellt?	6
2.3 Verschiedene Testarten	11
2.4 Verschiedene Test-Aufgabentypen	14
2.5 Testdurchführung	18
2.6 Testauswertung	20
3 Rechnerunterstützte Tests	36
3.1 Durchführung von rechnerunterstützten Tests	36
3.2 Vorteile von rechnerunterstützten Tests	37
3.3 Einschränkungen von rechnerunterstützten Tests	40
3.4 Eignung verschiedener Aufgabentypen für rechnerunterstützte Tests	41
3.5 Datenbankgestützte Testumgebungen	44
4 Anforderungsanalyse	46
4.1 Testarten	46
4.2 Aufgabentypen	46
4.3 Werkzeug zur Test- und Aufgabenadministration	47
4.4 Werkzeug zur Testdurchführung	51
4.5 Werkzeug zur Auswertung	51
4.6 Anforderungen zur Sicherheit	53
5 Der Datenbankentwurf	54
5.1 Das Entity-Relationship-Modell	54
5.2 Die Relationen	64
5.3 Durchgeführte Optimierungen	73
5.4 Konsistenzerhaltung in der Datenbank	76

6 Entwurf der Werkzeuge	83
6.1 Testdurchführung	83
6.2 Auswertung	90
7 Die SQL-Anfragen	95
7.1 Anfragen für das Werkzeug zur Testdurchführung (WETED)	95
7.2 Anfragen für das Werkzeug zur Auswertung (WEBEWAU)	97
8 Erfahrungen bei der Durchführung von rechnerunterstützten Tests	105
8.1 Informationen über die Veranstaltung	105
8.2 Ablauf der Testdurchführung	106
8.3 Technische Durchführung	106
8.4 Ergebnisse	107
9 Zusammenfassung und Ausblick	111
10 Literaturverzeichnis	114

Abbildungsverzeichnis

1.1	Mögliche Architektur einer datenbankgestützten Lernumgebung über das World Wide Web.	2
2.1	Beispiel einer Multiple-Choice-Aufgabe mit Mehrfachnennung	15
2.2	Beispiel einer Multiple-Choice-Aufgabe ohne Mehrfachnennung	15
2.3	Beispiel einer Zuordnungsaufgabe	16
2.4	Beispiel einer Umordnungsaufgabe	16
2.5	Beispiel einer Ergänzungsaufgabe	17
2.6	Beispiel einer Kurzaufsatz-Aufgabe	17
2.7	CBAA Dreieck (Computer-Based Alternative Assessment Dreieck)	22
2.8	Schwierigkeitsindex von Aufgaben verschiedener Schwierigkeit	32
2.9	Zusammenhang zwischen Schwierigkeitsindex und Trennschärfekoeffizient	35
5.1	Das ER-Modell der Aufgabe	55
5.2	Das ER-Modell des Aufgabengebiets	57
5.3	Das ER-Modell des Tests	58
5.4	Das ER-Modell der Testinstanz	60
5.5	Das ER-Modell, mit dem alle Teilnehmenden eines Testdurchlaufs die gleichen Aufgaben erhalten	62
5.6	Das ER-Modell der Datenbank	63
5.7	Verhalten der Datenbank beim Einfügen und Löschen von Beziehungen zwischen Aufgabengebieten	75
6.1	Die Oberfläche zur Testdurchführung im WETED	86
6.2	Die Oberfläche zur Einsicht im WETED	87
6.3	Die Oberfläche zur Bewertung der Ergebnisse im WEBEWAU	91
6.4	Die Oberfläche zur Aufgabenanalyse im WEBEWAU	92
6.5	Die Oberfläche der Testanalyse im WEBEWAU	93
7.1	SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets	95
7.2	SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets unter Ausschluß vorher bearbeiteter Aufgaben	96

7.3	SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets, die bereits von Teilnehmenden bearbeitet wurden	96
7.4	SQL-Anfrage zur Bestimmung des Zeitpunktes aller Antworten eines Teilnehmenden zu einer Aufgabe sowie der Qualität der Antwort	97
7.5	SQL-Anfrage, die den Rohwert von <i>Teilnehmer1</i> in der Testinstanz <i>Testinstanz1</i> mit der Testnummer <i>Test1</i> bestimmt	97
7.6	SQL-Anfrage zur Bestimmung aller Testinstanzen, bei denen eine Bewertung vorgenommen werden kann	98
7.7	SQL-Anfrage zur Bestimmung aller Aufgaben für die Aufgabenanalyse aufgrund vorgegebenem Suchstring und Testinstanz (TestNr und TINr)	98
7.8	Berechnung des Schwierigkeitsindizes für eine Aufgabe (AufgabenNr) bezogen auf eine Testinstanz (TestNr und TINr)	99
7.9	Berechnung der Standardabweichung des Rohwerts, des durchschnittlichen Rohwerts und der Anzahl der Teilnehmenden	99
7.10	Berechnung der Standardabweichung der Punktzahl und der durchschnittlichen Punktzahl einer Aufgabe	100
7.11	Bestimmung der Trennschärfe	100
7.12	Anzahl der Nennungen der Items einer Aufgabe	101
7.13	Bestimmung eines Teils der Formel zur Berechnung der Trennschärfe (Item gewählt)	102
7.14	Bestimmung eines Teils der Formel zur Berechnung der Trennschärfe (Item nicht gewählt)	102
7.15	Auswahl der Aufgaben zur Testhalbierung	103
7.16	Bestimmung der Korrelation zweier Testhälften	104
8.1	Umfrage-Aufgabe	107
8.2	Aufgabe für die Analyse	110

Tabellenverzeichnis

2.1	Spektrum der Notenverteilung bei der Bewertung des gleichen Deutschaufsatzes durch 42 Fachlehrer	7
2.2	Die Zuordnung der Aufgabentypen zu den Testarten	19
3.1	Eignung verschiedener Aufgabentypen für rechnerunterstützte Tests	42
5.1	Attribute der Entity-Mengen des ER-Modells Aufgabe	56
5.2	Attribute der Entity-Mengen des ER-Modells Aufgabengebiet	57
5.3	Attribute der Entity-Mengen des ER-Modells Test	59
5.4	Attribute der Entity-Mengen des ER-Modells Test	61
5.5	Attribute von WWWT_Aufgabe	64
5.6	Attribute von WWWT_Schw_Stufe	65
5.7	Attribute von WWWT_Item	65
5.8	Attribute von WWWT_Loesungswort	65
5.9	Attribute von WWWT_AG	66
5.10	Attribute von WWWT_Au_liegtin_AG	66
5.11	Attribute von WWWT_AG_liegtin_AG	66
5.12	Attribute von WWWT_Test	67
5.13	Attribute von WWWT_Anleitung	67
5.14	Attribute von WWWT_GenT_enth_AG	68
5.15	Attribute von WWWT_CAT_enth_AG	68
5.16	Attribute von WWWT_CAT_kann_verw	68
5.17	Attribute von WWWT_Gewichtung	69
5.18	Attribute von WWWT_Gew_fuer_SS	69
5.19	Attribute von WWWT_Teilnehmende	69
5.20	Attribute von WWWT_Testinstanz	70
5.21	Attribute von WWWT_Bewertungsart	70
5.22	Attribute von WWWT_Note	71
5.23	Attribute von WWWT_Notenschl	71
5.24	Attribute von WWWT_Teilnahme	71
5.25	Attribute von WWWT_BearbeitungsZ	72
5.26	Attribute von WWWT_Testprotokoll	72
5.27	Attribute von WWWT_Gew_Loesung	73
5.28	Attribute von WWWT_Aufg_vorgabe	73

8.1	Verteilung der Schwierigkeitsindizes	108
8.2	Verteilung der Trennschärfekoeffizienten	109

Vorbemerkung zur Gleichstellung der Geschlechter

Die Gleichstellung zwischen Frau und Mann in der Gesellschaft sollte heutzutage selbstverständlich sein. Leider wird dies in der traditionellen Form zur Bezeichnung unbestimmter Personen in der deutschen Sprache nicht berücksichtigt. Es wird stets die maskuline Form verwendet. “Der Prüfer fragt die Teilnehmer”.

Verschiedene Alternativen hierzu sind nach [Lud92]:

1. Die *vollständige Aufzählung*. “Der Prüfer oder die Prüferin fragt die Teilnehmer und Teilnehmerinnen.”
2. die *Mischschreibweise*. “Der/Die PrüferIn fragt die TeilnehmerInnen.”
3. Die *Diskriminierung der Männer*. “Die Prüferin fragt die Teilnehmerinnen.”
4. Die *gemischten Prototypen*. “Die Prüferin, nennen wir sie Kerstin, fragt die Teilnehmer, nennen wir sie Klaus und Werner.”
5. Die *Verwendung des Neutrums*. “Das Prüfer fragt die Teilnehmer.”

Dabei sind die Alternativen eins, zwei und vier für die Lesenden schwerer zu verstehen, da die Sätze in die Länge gezogen werden, ohne mehr Inhalt zu vermitteln. Sie sollten deshalb bei wissenschaftlichen Arbeiten vermieden werden. Die dritte Alternative ist im Prinzip genauso schlecht wie die herkömmliche Schreibweise, sie dreht die Diskriminierung einfach um. Die fünfte Alternative verwirrt die Lesenden zu sehr, wie Ludewig nach Anwendung dieser Möglichkeit zu berichten weiß.

Das Beibehalten der traditionellen Form, mit der Begründung, daß keine brauchbaren Alternativen verfügbar sind, ist allerdings unbefriedigend.

Deshalb werde ich für diese Ausarbeitung bei der Mehrzahl, soweit möglich, eine geschlechtsunspezifische Benennung wählen. Statt “die Teilnehmer” oder “die Teilnehmerinnen” also “die Teilnehmenden”.

Dies ist beim Verwenden des Singulars allerdings nicht möglich. In diesem Fall werde ich abwechselnd beide Geschlechtsformen wählen, ohne jedoch wie in Alternative vier Prototypen zu benutzen. Die Geschlechtsform wird dabei für die gleiche Funktion im gleichen Abschnitt beibehalten. Es wird also beispielsweise im gleichen Abschnitt immer von der Prüferin und dem Teilnehmer die Rede sein, obwohl beide Rollen auch vom jeweils anderen Geschlecht besetzt sein können.

Kapitel 1

Einleitung

Einführung in das Themengebiet

Traditionelle Lehrveranstaltungen sind dadurch gekennzeichnet, daß sich die Lehrenden und Lernenden zur gleichen Zeit am selben Ort einfinden müssen.

Es gibt bereits etablierte Ansätze, die diesen Rahmen verlassen haben. Dazu zählen Schulfunk und Schulfernsehen. Diese Lehrformen bedienen sich eines (für Lehrveranstaltungen) neuen Mediums (Funk bzw. Fernsehen) und trennen so die zeitliche und räumliche Bindung zwischen Lehrenden und Lernenden. Ein Nachteil dieser Lehrformen ist allerdings die fehlende Interaktion zwischen Lehrenden und Lernenden.

Durch die Bereitstellung eines weltweiten Rechner-Netzwerkes - des Internets - als neuem Medium ist es möglich, Lehrveranstaltungen räumlich getrennt abzuhalten und trotzdem eine Interaktion zwischen den Lehrenden und Lernenden zu ermöglichen. Die Vorteile dieser Lehrveranstaltungsform werden im folgenden kurz exemplarisch an Hand der Hochschule skizziert.

Lernenden kann durch Lehrveranstaltungen, die über das Internet gehalten werden, eine größere Auswahl an verschiedenen Themengebieten angeboten werden, die ansonsten gerade an kleinen Hochschulen oft eingeschränkt ist. Die Wahrscheinlichkeit des Absagens von Veranstaltungen wegen einer zu geringen Anzahl der Teilnehmenden wird durch die Ausweitung der Anzahl der potentiellen Teilnehmenden verringert. Überfüllte Hörsäle können durch das Abhalten von Veranstaltungen über das Internet vermieden werden. Die Teilnehmenden können diese Veranstaltungen mit privaten Internetanschlüssen von zu Hause mitverfolgen.

Ein weiterer Vorteil des Internet-basierten Lehrens ist ein ökonomischer. Während in der heutigen Situation die Hochschulen aufgrund von politisch vorgegebenen Sparmaßnahmen das Lehrangebot einschränken müssen, kann dem durch ein zusätzliches Angebot über das Internet langfristig entgegengesteuert werden. So muß eine Hochschule nicht mehr jede Veranstaltung selber anbieten und kann so Mittel einsparen, ohne das Angebot für die Lehrenden einzuschränken. Hohe Investitionskosten zur Einführung entfallen bei den Hochschulen, da die Infrastruktur, wie Rechnerpools mit Internetanschluß, bereits vorhanden

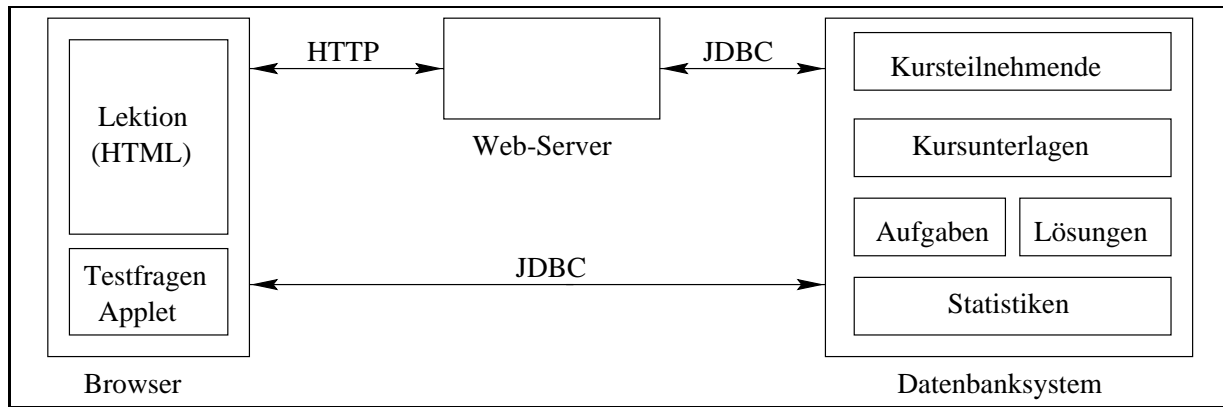


Abbildung 1.1: Mögliche Architektur einer datenbankgestützten Lernumgebung über das World Wide Web.

ist.

Eine genauere Analyse über das rechnergestützte Lehren und Lernen findet sich in [Otto95]. Ein Einsatz von Lehrveranstaltungen über das Internet innerhalb der Ausbildung ist allerdings nur dann sinnvoll, wenn neben dem Lehren auch die Möglichkeit zur Durchführung von Tests besteht. Auf diese Weise kann der Wissensstand der Lernenden beurteilt werden. Dabei müssen prinzipiell verschiedene Arten von Tests unterschieden werden. Es existieren Eingangstests, Tests zur Lernfortschrittskontrolle sowie Abschlußtests, die sich durch verschiedene Charakteristiken auszeichnen.

Zur Durchführung von Tests (wie auch Lehrveranstaltungen) im Internet bietet sich besonders das *World Wide Web* an, da die Inhalte leicht graphisch aufbereitet werden können und entsprechende Browser plattformunabhängig zur Verfügung stehen.

Eine *datenbankgestützte Lernumgebung* ermöglicht es, neben den unterschiedlichen Kursunterlagen, sämtliche Prüfungsfragen, Notenskalen und auch die Testergebnisse der einzelnen Teilnehmenden zu verwalten und Möglichkeiten für die statistische Auswertung bereitzustellen.

In Abbildung 1.1 ist eine mögliche Architektur einer solchen Lernumgebung dargestellt. Die Lektionen einer Lehrinheit, wie auch die Testfragen, werden über einen WWW-Browser dargestellt. Die Darstellung der Lektionen wird über HTML (Hypertext Markup Language) definiert, wohingegen die Testfragen in einem Java-Applet angezeigt werden. Um Informationen zur Darstellung der Lektionen zu erhalten, kommuniziert der Browser über HTTP (Hypertext Transfer Protocol) mit einem Web-Server. Dieser erhält die Kursunterlagen für die Lektionen aus dem Datenbanksystem über die Java-eigene Datenbankschnittstelle JDBC (Java Database Connectivity). Das Testfragen-Applet kommuniziert über JDBC direkt mit dem Datenbanksystem, um die entsprechenden Testfragen zu erhalten. Im Datenbanksystem sind die Kursunterlagen, die Aufgaben mit Lösungen, sowie Informationen zu den am Kurs Teilnehmenden und statistische Informationen enthalten.

Aufgabenstellung

Innerhalb dieser Arbeit werden zuerst die Grundlagen der pädagogischen Psychologie der Durchführung und Bewertung von Tests erarbeitet. Auf dieser Basis wird ein geeignetes Datenbankschema entworfen, das in der Lage ist, verschiedene Typen von Testfragen mit unterschiedlichem Schwierigkeitsgrad und unterschiedlichem Typ (Multiple Choice Test, Lückentest) und deren Antworten sowie den erzielten Testergebnissen zu speichern. Anschließend wird ein Java-basierter Testfragen-Generator sowie ein Werkzeug zur Auswertung von Statistiken entworfen, die in der Lage sein sollten, für die Studierenden aufgrund einer Zufallsverteilung Testfragen aus einem Fragenkatalog auszuwählen, die Antworten zu protokollieren und den erzielten Lernerfolg zu bewerten. Den Abschluß bildet eine Diskussion der rechtlichen Fragestellungen.

Gliederung der Arbeit

Das folgende Kapitel gewährt einen Einblick in die Grundlagen der Durchführung und Bewertung von Tests. Dabei werden die Anforderungen an Tests beschrieben sowie die verschiedenen Test- und Aufgabenarten vorgestellt. Anschließend erfolgt eine Beschreibung der Testdurchführung sowie der Auswertungsmöglichkeiten.

Thema des dritten Kapitels sind die Besonderheiten rechnerunterstützter Tests. Es werden die Vor- und Nachteile erläutert und die Eignung verschiedener Aufgabentypen für das rechnerunterstützte Testen analysiert. Abschließend folgt eine kurze Diskussion der Vorteile datenbankgestützter Testumgebungen.

Kapitel 4 enthält die Spezifikation der Anforderungen an die für einen rechnerunterstützten Testbetrieb benötigten Werkzeuge. Dies sind im einzelnen ein Werkzeug zur Test- und Aufgabenadministration, ein weiteres zu Durchführung und schließlich ein Werkzeug zur Auswertung von Tests.

Kapitel 5 beinhaltet den Datenbankentwurf für eine rechnerunterstützte Testumgebung. Neben dem Entity-Relationship-Modell und den sich daraus ergebenden Relationen wird auf die Konsistenzerhaltung innerhalb der Datenbank eingegangen.

In Kapitel 6 wird der Entwurf der Werkzeuge zur Testdurchführung und Auswertung vorgestellt. Ein Entwurf des Werkzeugs zur Test- und Aufgabenadministration entfällt, da die wesentlichen Anforderungen bereits im Datenbankentwurf berücksichtigt wurden.

Inhalt des siebten Kapitels stellen die SQL-Anfragen dar, die von den entworfenen Werkzeugen verwendet werden.

In Kapitel 8 folgt eine Beschreibung der Erfahrungen, die bei der praktischen Durchführung von rechnerunterstützten Tests im Rahmen eines Hauptseminars *Multimediale Anwendungen im Internet* gewonnen wurden.

Kapitel 9 schließt die Arbeit mit einer Zusammenfassung der Ergebnisse und einem Ausblick auf die weitere mögliche Entwicklung ab.

Kapitel 2

Grundlagen der pädagogischen Psychologie zur Durchführung und Bewertung von Tests

In diesem Kapitel werden die Grundlagen der pädagogischen Psychologie zur Durchführung und Bewertung von Tests erarbeitet. Dazu wird zuerst der Begriff Test genauer erläutert. Anschließend werden die Anforderungen spezifiziert, die an einen Test gestellt werden. Es folgt die Vorstellung verschiedener Testarten. Desweiteren wird auf die Testdurchführung eingegangen. Zum Schluß werden die Möglichkeiten der Testauswertung besprochen.

2.1 Was ist ein Test?

Das Wort “*Test*” stammt aus dem Englischen und hat die Bedeutung von Probe, Prüfung, psychologisches Experiment oder Untersuchung [Duden91].

In der Psychologie hat das Wort “Test” mehrerer Bedeutungen. Es steht für

- ein Verfahren zur Untersuchung eines *Persönlichkeitsmerkmals*, wie beispielsweise die Intelligenz oder die Fähigkeit, bestimmte Mathematikaufgaben zu lösen,
- den Vorgang der Durchführung einer Untersuchung,
- die Gesamtheit der zu einer Durchführung notwendigen Requisiten,
- jede Untersuchung, sofern sie Stichprobencharakter hat,
- gewisse mathematisch-statistische Prüfverfahren,
- kurze, außerplanmäßige “Zettelarbeiten” im Schulunterricht [LiRa94].

Dabei ist von den eben genannten Bedeutungen die erste am wichtigsten, da diese Bedeutung auch im folgenden bei dem Gebrauch des Wortes Test in dieser Arbeit verwendet wird. Es folgt eine genaue Definition des Begriffes aufgrund dieser Bedeutung.

Definition: Ein Test ist ein *systematisches Verfahren* zur Messung¹ einer Stichprobe des Verhaltens eines Menschen, um dieses Verhalten anhand von Maßstäben oder Normen zu bewerten.[GaBe86]

Mit einem *systematischen Verfahren* ist ein standardisiertes Verfahren gemeint, das allen zu Testenden die gleichen Rahmenbedingungen bietet. Diese müssen detailliert festgelegt sein. Dazu zählen beispielsweise die Zeit, in der eine Aufgabe bewältigt werden muß oder genaue Aufgabenstellungen. Aber auch Erläuterungen, die vor oder während der Bearbeitung des Tests von einer Aufsicht gegeben werden, müssen genau festgelegt sein.

Ein Test soll bei der zu testenden Person ein bestimmtes Verhalten auslösen (z.B. ein Kreuz hinter eine mögliche Antwort machen). Lediglich dieses Verhalten kann beobachtet und damit gemessen werden. Die Art und Weise, wie dieses Ergebnis erzielt wurde, kann nicht beobachtet und damit auch nicht gemessen werden, z.B. die Gedanken beim Lösen der Aufgaben.

Es ist zu beachten, daß ein Test immer nur die Erhebung einer *Verhaltensstichprobe* darstellt. Ein schlechtes Abschneiden eines Testkandidaten bei einem Test kann also auch an der Qualität des Stichprobenverfahrens liegen.

Ein Test mißt eine Stichprobe des menschlichen Verhaltens. In einem *Bewertungsprozeß* muß festgelegt werden, wie dieses Ergebnis zu bewerten ist. Dazu gibt es zwei prinzipielle Vorgehensweisen. Bei Anwendung einer normbezogenen Bewertung werden die Testergebnisse der Testkandidaten direkt miteinander verglichen und daraufhin bewertet. Im Gegensatz dazu wird bei einer kriteriumsbezogenen Bewertung das Ergebnis eines Tests unabhängig von den anderen Testkandidaten bewertet. Die Kriterien der Bewertung werden im voraus festgelegt. Beispielsweise erhält Studentin A bei einem Test 75 von 100 Punkten, Studentin B beim gleichen Test 60 Punkte und Studentin C lediglich 45 Punkte. Bei der normbezogenen Bewertung kann gesagt werden, daß Studentin A in dem von diesem Test gemessenen Kriterium besser ist als Studentin B, und beide besser als Studentin C sind. Studentin A würde beispielsweise als Note eine Eins, Studentin B eine Zwei und Studentin C eine Drei erhalten. Bei einer kriteriumsbezogenen Bewertung wird die Bewertung unabhängig vom Abschneiden der Testkandidaten festgelegt. Bei dem obigen Test wurde z.B. festgelegt, daß der Test erst mit 80 oder mehr Punkten als bestanden gilt. In diesem Fall hätte keine der Studierenden den Test bestanden.

¹Ein naturwissenschaftlich geprägter Mensch mag sich über den Begriff "Messung" in diesem Zusammenhang wundern, da keine physikalische Größe gemessen wird. Es können auch nicht-physikalische Größen gemessen werden, wobei hier in der Regel nicht die Exaktheit einer naturwissenschaftlichen Messung erreicht werden kann.

2.2 Welche Anforderungen werden an einen Test gestellt?

Es gibt verschiedene *Gütekriterien*, die einen Test auszeichnen. Sollte ein Test diese Kriterien nicht, oder nur mangelhaft erfüllen, werden die aus diesem Test gewonnenen Resultate ebenso ungenügend sein.

Die Hauptanforderungen an einen Test sind:

- die Objektivität,
- die Reliabilität,
- und die Validität.

Neben diesen Hauptanforderungen, die auf jeden Fall erfüllt sein sollen, gibt es weitere Anforderungen an einen Test. So soll ein Test

- normiert,
- vergleichbar,
- ökonomisch,
- und nützlich sein.

Die Kriterien sind entnommen aus [LiRa94]. Es folgt eine Erläuterung der verschiedenen Kriterien.

2.2.1 Die Objektivität eines Tests

Die *Objektivität* eines Tests verlangt, daß alle subjektiven Faktoren von Seiten der Untersuchenden ausgeschlossen werden. Dies bezieht sich sowohl auf die Durchführung, die Auswertung als auch auf die Interpretation eines Tests.

2.2.1.1 Durchführungsobjektivität

Bereits während der Durchführung des Tests ist darauf zu achten, daß alle Testkandidaten die gleichen Bedingungen zur Bearbeitung des Tests erhalten. Zu den dazu nötigen Voraussetzungen zählen nach Biglmaier [Big69] eine "eindeutige, ausführliche Testanleitung" sowie die Einhaltung genauer Regeln zur Durchführung (insbesondere die Zeitgrenzen und die Verweigerung zusätzlicher Lösungshilfen). Ebenso müssen die erlaubten Hilfsmittel (z.B. Taschenrechner, Formelsammlung ...) festgelegt werden. Werden diese Anforderungen bei einem Test nicht eingehalten, kann aus den Ergebnissen des Tests wenig bis gar nichts geschlossen werden.

Erhält ein Testkandidat eine genauere Testanleitung für einen Test als ein anderer, kann er den Test einfacher bearbeiten und so leichter eine höhere Punktzahl erzielen. Ein Beispiel

Note	Anzahl der Nennungen
sehr gut	1 mal
gut	5 mal
befriedigend bis gut	1 mal
voll befriedigend	1 mal
befriedigend	8 mal
noch befriedigend	3 mal
ausreichend	7 mal
nicht (ganz) ausreichend	3 mal
mangelhaft	11 mal
ungenügend	2 mal

Tabelle 2.1: Spektrum der Notenverteilung bei der Bewertung des gleichen Deutschaufsatzes durch 42 Fachlehrer (entnommen aus [Uls49])

hierfür wäre ein Multiple-Choice Test mit verschiedenen Arbeitsanleitungen, wobei in einer dieser Anleitungen beschrieben wird, daß keine Mehrfachnennungen möglich sind, andere Anleitungen diese Information dagegen nicht erhalten. Die Kandidaten, die diese zusätzliche Information nicht erhalten, können im Test mehr Fehler machen, da sie im Zweifelsfall mehrere mögliche Antworten ankreuzen, was auf jeden Fall zu einer fehlerhaften Antwort führt.

Wurde die oben genannte Information in der (einzigen) Anleitung nicht erwähnt, darf sie auch nicht einigen Testkandidaten auf Anfrage während der Durchführung des Tests von der Aufsicht gegeben werden, da auch dies die Objektivität einschränken würde. Gleiches gilt für verschiedene Zeitgrenzen sowie Hilfsmittel bei dem gleichen Test.

2.2.1.2 Auswertungsobjektivität

Auswertungsobjektivität bedeutet, daß der von einem Testkandidaten ausgeführte Test bei verschiedenen, auswertenden Personen immer dasselbe Ergebnis liefern sollte.

Dies ist bei einfachen Aufgabentypen, die lediglich “Ja” oder “Nein” als Antwort zulassen, einfach zu verwirklichen. Anders bei komplexen Antworten:

In einer von Robert Ulshöfer initiierten Untersuchung ließ er einen Deutschaufsatz von 42 Fachlehrern zensieren. In Tabelle 2.1 sind die verschiedenen Notenvorschläge der an der Untersuchung teilnehmenden Lehrer dargestellt. Dieses Ergebnis zeigt, daß die Auswertung des Aufsatzes keineswegs objektiv war.

Die Auswertungsobjektivität kann bei freien Antworten erhöht werden, indem den Auswertenden genau eingegrenzte richtige Lösungen sowie Kriterien für die Bewertung zur Verfügung gestellt werden.

2.2.1.3 Interpretationsobjektivität

Die *Interpretationsobjektivität* betrifft die Unabhängigkeit der Interpretation der Testergebnisse von der interpretierenden Person, die nicht mit der auswertenden Person übereinstimmen muß. Sie ist gegeben, wenn aus den gleichen Auswertungsergebnissen (verschiedener Testkandidaten) die gleichen Schlüsse gezogen werden. In einem normierten Test, der als Auswertungsergebnis einen numerischen Wert liefert und so die Position des Testkandidaten auf einer Testskala bestimmt, ist die Interpretationsobjektivität erreicht.

2.2.2 Die Reliabilität eines Tests

Unter der *Reliabilität* eines Tests wird die Zuverlässigkeit verstanden, mit der ein Test ein bestimmtes Persönlichkeits- oder Verhaltensmerkmal mißt. Dabei ist nicht von Bedeutung, ob der entsprechende Test dieses Merkmal messen soll. Dies ist eine Frage der Validität. Eine vollkommene Reliabilität ist gegeben, wenn die Testergebnisse genau reproduzierbar sind, falls sie unter den gleichen Bedingungen gewonnenen wurden.

Um ein Maß der Reliabilität anzugeben, kann der *Reliabilitätskoeffizient* bestimmt werden. Dieser gibt an, inwieweit die Testergebnisse unter gleichen Bedingungen reproduzierbar sind.

Es gibt verschiedene Methoden, diesen Reliabilitätskoeffizienten zu bestimmen. Je nach Methode werden verschiedene Reliabilitäts-Formen unterschieden.

2.2.2.1 Die Test-Retest-Reliabilität

Der Reliabilitätskoeffizient der *Test-Retest-Reliabilität* wird folgendermaßen bestimmt: Der Test wird von denselben Testkandidaten zweimal durchgeführt. Die Korrelation der Rangplätze der Testkandidaten beider Durchläufe ergibt den Reliabilitätskoeffizienten.

2.2.2.2 Die Paralleltest-Reliabilität

Zur Messung der *Paralleltest-Reliabilität* wird zusätzlich zum Test ein weiterer Test - der Paralleltest - benötigt. Dieser muß sowohl das gleiche Persönlichkeitsmerkmal messen als auch die gleiche Punktverteilung für die Ausprägung des zu messenden Merkmals vergeben. Diese beiden vergleichbaren Tests werden von den gleichen Testkandidaten bearbeitet. Die Korrelation der Punktzahl der einzelnen Testkandidaten bestimmt den Reliabilitätskoeffizienten.

2.2.2.3 Die innere Konsistenz

Die *innere Konsistenz* mißt die Reliabilität eines Tests in lediglich einem Testdurchlauf. Sie ist damit unabhängig vom möglichen Wissenszuwachs einzelner Testkandidaten zwischen zwei Vergleichstests. Zudem ist dieses Meßverfahren praxisnah, da sich nicht genau die

gleichen Testkandidaten für eine zweite Testdurchführung einfinden müssen.
Zur Messung der inneren Konsistenz werden zwei Methoden unterschieden:

Die Testhalbierung Bei der Testhalbierungs-Methode (auch Split-half-Methode) wird ein Test von einer Menge von Testkandidaten einmal ausgeführt. Anschließend werden die einzelnen Aufgaben des Tests in zwei vom Aufgabentyp, von der Punktverteilung und vom Schwierigkeitsgrad gleichwertige Hälften geteilt. Diese Teile werden für jeden Testkandidaten gesondert ausgewertet und die Korrelation zwischen diesen Ergebnissen ergibt den Reliabilitätskoeffizienten.

Die Konsistenzanalyse Die Konsistenzanalyse kann als eine Verallgemeinerung der Testhalbierungs-Methode gesehen werden. Dabei wird der Test nicht nur in zwei Hälften aufgeteilt sondern jedes Test-Element (Aufgabe) einzeln betrachtet. Die Reliabilität wird über bestimmte Kennwerte, wie der Schwierigkeitsgrad dieser Test-Elemente, ermittelt und nicht durch die Korrelation verschiedener Testwerte-Paare, wie in den obigen Verfahren. Eine genaue Beschreibung des Verfahrens ist in [LiRa94] gegeben. Es ist anzumerken, daß bei diesem Verfahren davon ausgegangen wird, daß jedes Test-Element das gleiche Merkmal mißt und nicht verschiedene Test-Elemente verschiedene Aspekte des zu messenden Merkmals.

Der Reliabilitätskoeffizient liegt zwischen 1,0 und 0,0, wobei 1,0 eine vollständige Reliabilität bedeuten würde. Eine Reliabilität von weniger als 0,8 ist “generell nicht akzeptabel, wenn die Testinformation dazu verwendet werden soll, wichtige Entscheidungen über Individuen zu treffen” [GaBe86]. Wird der Reliabilitätskoeffizient für den gleichen Test mit allen oben genannten Verfahren gemessen, wird es in der Regel zu Abweichungen zwischen den einzelnen Werten kommen, die aus den Methoden eigenen Meßfehlerarten und -anteilen resultieren.

2.2.3 Die Validität eines Tests

Unter der *Validität* eines Tests wird die *Gültigkeit* eines Tests verstanden, das bedeutet, daß ein Test die Merkmale mißt, für die der Test vorgesehen ist. Es existieren verschiedene Arten der Validität, die als Inhaltsvalidität, Konstruktvalidität und Kriteriumsvalidität bezeichnet werden.

2.2.3.1 Die inhaltliche Validität

Die *inhaltliche Validität* bedeutet, daß die Testfragen inhaltlich eine repräsentative Auswahl des zu untersuchenden Merkmals darstellen.

Sollte beispielsweise eine Unterrichtslektion getestet werden, so muß der entsprechende Test die Lernziele des Unterrichtsstoffes abdecken.

Die inhaltliche Validität eines Tests wird in der Regel von Experten des jeweiligen zu testenden Merkmals bestimmt.

2.2.3.2 Die Konstruktvalidität

Die *Konstruktvalidität* bestimmt den Grad, in dem ein Test das vorgegebene Merkmal mißt. Diese Validität bezieht sich weniger auf den Inhalt der Testfragen sondern mehr auf die Art der Fragen und die Durchführung eines Tests.

So bestimmt ein Test, dessen Zeitrahmen zu knapp bemessen ist, mehr die Geschwindigkeit, in der die Testkandidaten die Aufgabenstellung erfassen und lösen. Die eigentlich durch den Test zu überprüfenden Merkmale werden in den Hintergrund gedrängt. Dabei ist es denkbar, Tests zu konstruieren, die genau diese Fähigkeit der schnellen Aufgabenerfassung und -lösung messen sollen.

Die Konstruktvalidität kann durch die Korrelation mit den tatsächlich vorhandenen Fähigkeiten der Testkandidaten in dem zu messenden Merkmal bestimmt werden.

Wird beispielsweise in der Schule ein Mathematik-Test durchgeführt, sollten die Schülerinnen, die sich im Unterricht durch gute Leistungen besonders hervorgehoben haben, auch in diesem Test gute Leistungen zeigen. Es kann nicht erwartet werden, daß im Unterricht unauffällige Schülerinnen schlecht abschneiden, da dieses Verhalten nicht am Unverständnis des Unterrichtsstoffes liegen muß.

2.2.3.3 Die Kriteriumsvalidität

Bei der *Kriteriumsvalidität* wird der Grad der Validität über die Korrelation zu einem Außenkriterium gewonnen. Dieses Außenkriterium wird für die Testkandidaten unabhängig vom Test bestimmt. Dabei kann das Außenkriterium selbst das vom Test zu messende Merkmal sein oder eine hinreichende Validität zu dem zu messenden Merkmal besitzen.

Wird beispielsweise ein Test entworfen, der aus den Teilnehmenden eines Anfängerkurses einige für einen fortgeschrittenen Kurs auswählen soll, so müssen zur Bestimmung der Kriteriumsvalidität zunächst alle Teilnehmenden in den fortgeschrittenen Kurs übernommen werden. Als Außenkriterium wird der tatsächliche Erfolg der Teilnehmenden in dem fortgeschrittenen Kurs gewählt. Sollten die durch den Test bestimmten Teilnehmenden Erfolg, und die anderen Teilnehmenden Mißerfolg im Kurs haben, war der Test kriteriumsvalide.

2.2.4 Die Normierung eines Tests

Die *Normierung* eines Tests dient dem Vergleich der Ergebnisse verschiedener Tests (, die das gleiche messen). Dazu muß ein Bezugssystem vorliegen, das es erlaubt, die einzelnen Testergebnisse der Kandidaten diesem Bezugssystem zuzuordnen. Für jeden normierten Test muß angegeben werden, wie die Testergebnisse dem Bezugssystem zuzuordnen sind. Als Beispiel für ein Bezugssystem kann der Intelligenzquotient gesehen werden. Für einen normierten IQ-Test muß angegeben werden, bei welchem Testergebnis welcher IQ-Wert zugewiesen wird (z.B. 70 Prozent der Aufgaben richtig gelöst ergibt einen IQ von 100 Prozent).

Die Normierung eines Tests bezieht sich immer auf eine Vergleichsgruppe. Diese kann entweder die Gesamtpopulation eines bestimmten geographischen Gebiets oder lediglich die

Population einer bestimmten sozialen Gruppe sein (z.B. Informatikstudenten im Hauptstudium).

2.2.5 Die Vergleichbarkeit eines Tests

Die *Vergleichbarkeit* eines Tests ist gegeben, wenn mindestens ein Paralleltest vorliegt oder Tests mit einer ähnlichen Validität vorhanden sind.

Die Vergleichbarkeit eines Tests ist von Bedeutung, falls zu unterschiedlichen Terminen die gleichen Merkmale gemessen werden sollen. Dabei kann in bestimmten Fällen nicht derselbe Test für zeitlich versetzte Messungen gewählt werden. So kann es vorkommen, daß die gleichen Testkandidaten an mehreren Terminen teilnehmen (beispielsweise durch das Wiederholen der gleichen Klassenstufe in der Schule oder bei Tests zur Überprüfung des Lernfortschritts). Auch könnten Informationen über den Test von den Testkandidaten, die den Test bereits zu einem früheren Termin bearbeitet haben, an die Testkandidaten, die den Test noch bearbeiten müssen, weitergegeben werden. In diesen Fällen ist es erforderlich, einen anderen Test zu benutzen. Ansonsten sind die Ergebnisse der einzelnen Testkandidaten nicht mehr vergleichbar, da die Ausgangsbedingungen der einzelnen Kandidaten unterschiedlich sind. Einige Testkandidaten haben sich auf die Fragen bereits vor dem Test vorbereiten können, während andere erst im Laufe des Tests mit diesen Fragen konfrontiert werden.

2.2.6 Die Ökonomie eines Tests

Die Merkmale eines *ökonomischen* Tests sind eine kurze Durchführungszeit, der geringe Materialverbrauch, eine einfache Handhabung, die Ausführung als Gruppentest sowie die Möglichkeit einer einfachen und schnellen Auswertung.

Die Vorteile eines ökonomischen Tests liegen auf der Hand. Sie verbrauchen weniger Ressourcen (wie Zeit, Raum und Materialien), die so eingespart (z.B. Materialien) oder für andere Zwecke benutzt werden können (z.B. Zeit).

2.2.7 Die Nützlichkeit eines Tests

Nützlich ist ein Test, wenn er ein Merkmal mißt, für dessen Untersuchung ein praktisches Bedürfnis besteht.

Ein Test ist kaum nützlich, falls es bereits eine Vielzahl anderer Tests gibt, die das gleiche Merkmal messen. Sehr nützlich ist er hingegen, wenn noch kein anderer Test dieses Merkmal messen kann.

2.3 Verschiedene Testarten

Tests können nach verschiedenen Kriterien eingeordnet werden. Es folgt eine Aufzählung der verschiedenen Kriterien und der entsprechenden Test-Kategorien.

Es kann zwischen Tests mit verschiedenen Graden der allgemeinen Anwendbarkeit unterschieden werden. Dabei erfolgt eine Einteilung in standardisierte und nichtstandardisierte Tests [LiRa94].

Standardisierte Tests werden von einer großen Anzahl von Testkandidaten durchgeführt und müssen wissenschaftlich entwickelt sowie hinsichtlich der in Abschnitt 2.2 beschriebenen Hauptkriterien überprüft worden sein. Sie werden unter Standardbedingungen abgehalten und sind normiert. Der Test für medizinische Studiengänge (TMS), der in Deutschland zusätzlich zur Abiturnote als Zulassungskriterium für ein Medizinstudium verwendet wird, stellt ein Beispiel für einen standardisierten Test dar.

Nichtstandardisierte Test werden in der Regel lediglich einmal an einer geringen Anzahl von Testkandidaten durchgeführt und genügen den Kriterien eines standardisierten Tests nicht. Zu den nichtstandardisierten Tests zählen fast alle an deutschen Schulen oder Hochschulen durchgeführten Tests.

Eine weitere Einteilung verschiedener Tests kann nach der Art des zu erfassenden Persönlichkeitsmerkmals erfolgen. Es wird nach Leistungs-, Fähigkeits- und Persönlichkeitstests unterschieden [MiNo95].

Leistungstests messen die motorische, sensorische oder kognitive Leistung des Testkandidaten.

Fähigkeitstests messen die Begabung der Testkandidaten. Zu den Fähigkeitstests zählen die Intelligenztests.

Persönlichkeitstests versuchen den Charakter, die Eigenschaften und Interessen der Testkandidaten zu bestimmen. Als Beispiel eines Persönlichkeitstest kann der Rorschach-Test genannt werden, bei dem den Testkandidaten zehn Bilder von Tintenklecksen vorgelegt werden, die diese interpretieren sollen.

Nach der Art der Durchführung können ebenfalls verschiedene Testarten festgelegt werden. Die folgenden Testarten können unterschieden werden: Befragungstests, Paper-Pencil-Tests sowie Materialbearbeitungstests [LiRa94].

Bei **Befragungstests** werden den Testkandidaten von anderen Personen vorgegebene Fragen gestellt, die diese mündlich beantworten. Die Befragungstests werden nach der Anzahl der Testkandidaten in **Gruppentests** mit mehreren Testkandidaten und **Einzeltests** mit lediglich einem gleichzeitig befragten Testkandidaten unterteilt [MiNo95].

Paper-Pencil-Tests sind eine weitverbreitete Testform, bei der die Fragen auf einem Bogen Papier festgehalten sind und mit einem Stift beantwortet werden.

Materialbearbeitungstests zeichnen sich dadurch aus, daß die Testkandidaten mit verschiedenen Materialien eine Aufgabe lösen müssen. Aus der Kindheit sollten jedem noch die verschiedenformigen Bauklötze bekannt sein, die durch ebenfalls verschieden geformte Öffnungen in einen Behälter gebracht werden sollten. Dies ist ein Beispiel für einen Materialbearbeitungstest.

Nach der Abhängigkeit vom Sprachverständnis werden verbale und nonverbale Tests un-

terschieden [LiRa94].

Bei **verbalen Tests** werden die einzelnen Aufgaben ausformuliert und in der Regel auf die gleiche Weise beantwortet, wohingegen bei **nonverbalen Tests** die Aufgaben als Schaubild oder auf andere nonverbale Weise gestellt werden und durch Ankreuzen oder ähnliche Verfahren beantwortet werden.

Die formale Struktur eines Test läßt eine Unterteilung in einfache und komplexe Tests zu [LiRa94].

Bei **einfachen Tests** ergibt jede Aufgabe einen Punktwert. Diese Punktwerte werden als Gesamtergebnis zu einem einzigen Punktwert zusammengezählt. **Komplexe Tests** dagegen erhalten bei der Auswertung mehrere Punktwerte, die zu einem vereinigt oder getrennt interpretiert werden können. Dies kann sich zum einen dadurch ergeben, daß verschiedene Aufgaben verschiedenen Punktwerten zugeordnet werden oder daß Aufgaben nach verschiedenen Gesichtspunkten bewertet werden. Beispielsweise kann eine Antwort zum einen inhaltlich und zum anderen nach der Rechtschreibung bewertet werden.

Der Bezugsgröße und dem Zweck eines Tests entsprechend wird zwischen normorientierten und kriteriumsorientierten Tests unterschieden [MiNo95].

Normorientierte Tests vergleichen das Ergebnis eines Testkandidaten mit einer Menge anderer Testresultate (der Norm). Bei diesen Tests soll das gemessene Merkmal der verschiedenen Testkandidaten möglichst genau differenziert werden.

Kriteriumsorientierte Tests prüfen, ob die Testkandidaten ein zuvor festgelegtes Kriterium erfüllen. Das Ergebnis eines Testkandidaten ist unabhängig von den Ergebnissen der anderen Testkandidaten.

Eine Unterscheidung verschiedener Tests nach der Art der Aufgabenbeantwortung kann ebenfalls erfolgen. Dabei wird nach der freien und der gebundenen Aufgabenbeantwortung unterteilt [LiRa94].

Tests mit gebundener Aufgabenbeantwortung erlauben lediglich die Auswahl aus verschiedenen vorgegebenen Antworten.

Tests mit freier Aufgabenbeantwortung verlangen vom Testkandidaten, selbst eine Antwort für die Aufgabe zu finden.

Die formale Struktur der Aufgaben läßt eine Aufteilung in hoch- und niedrigstrukturierte Tests zu [LiRa94].

Hochstrukturierte Tests enthalten nur Aufgaben, die lediglich eine richtige Antwort zulassen.

Niedrigstrukturierte Tests enthalten Aufgaben, bei denen die Antworten nicht mehr lediglich mit richtig oder falsch bewertet werden können.

Der Zeitpunkt der Testdurchführung relativ zu einer Lektion, die das zu messende Persönlichkeitsmerkmal der Testkandidaten schult, bietet ein weiteres Kriterium zur Unterscheidung verschiedener Testarten. Es kann eine Unterteilung in Eingangs- und Abschlußtests

sowie Tests zur Lernfortschrittskontrolle erfolgen.

Eingangstests messen das entsprechende Merkmal vor einer Schulung dieses Merkmals durch eine Vorlesung, eine Unterrichtseinheit, einen Kurs oder ähnlichem. Die Ergebnisse dieser Tests können dazu verwendet werden, die Lehrveranstaltung an die gemeinsamen Kenntnisse von allen Teilnehmenden anzupassen. Die dadurch eingesparte Zeit kann verwendet werden, um andere Aspekte genauer zu besprechen. Auch können durch Eingangstests Schwächen der Teilnehmenden aufgedeckt werden, die als Grundlagen der Lehrveranstaltung dienen. Diese können daraufhin gezielt in der Lehrveranstaltung erläutert werden.

Tests zur Lernfortschrittskontrolle werden zwischen einzelnen Einheiten einer Veranstaltung zur Schulung eines Persönlichkeitsmerkmals eingesetzt, um zu überprüfen, inwieweit der Stoff der Lehrveranstaltung von den Teilnehmenden aufgenommen wurde.

Abschlußtests werden nach einer Lehrveranstaltung durchgeführt. Mit ihnen wird überprüft, ob die Teilnehmenden der Lehrveranstaltung den durch die Lehrveranstaltung zu vermittelnden Stoff verstanden haben. In der Regel dienen Abschlußtests zur Benotung der Teilnehmenden. Die Mehrzahl der heutzutage an deutschen Schulen und Hochschulen durchgeführten Tests sind Abschlußtests. Sie haben den Nachteil, daß den Teilnehmenden das noch nicht vermittelte Wissen nicht mehr nahegebracht werden kann, da die Lehrveranstaltung bereits beendet ist. Die einzige verbleibende Möglichkeit ist, die Teilnehmenden die Lehrveranstaltung wiederholen zu lassen.

Weitere Testarten ohne spezielle Gegenstücke zur Einteilung sind der Eignungstest, der Schnelligkeitstest sowie der Niveautest [LiRa94].

Eignungstests stellen die prognostische Absicht in Hinblick auf einen Beruf in den Vordergrund des Interesses. Nach [PaDa36] darf dabei das Lernen und Üben eines Testkandidaten keine signifikante Änderungen seines Testergebnisses bewirken.

Schnelligkeitstest enthalten mehr Aufgaben, als ein Testkandidat in der vorgegebenen Zeit lösen kann. Dabei handelt es sich in der Regel um leichte bis mittelschwere Aufgaben. Es kommt also hauptsächlich auf die Geschwindigkeit der Testkandidaten an, in der sie eine Aufgabe lösen können.

Niveautests lassen den Testkandidaten viel Zeit für die Beantwortung der Fragen. Der Schwierigkeitsgrad nimmt aber von Frage zu Frage zu, so daß die letzten Fragen von kaum einem Teilnehmer gelöst werden können. Niveautests berücksichtigen kaum die Zeit, in der eine Aufgabe von einem Testkandidaten gelöst werden kann, und legen den Schwerpunkt auf das "geistige Niveau" [LiRa94] der Testkandidaten.

2.4 Verschiedene Test-Aufgabentypen

Neben den verschiedenen Testarten gibt es auch verschiedene Aufgabentypen, wobei einige Testarten bestimmte Aufgaben voraussetzen oder zumindest einige ausschließen.

Die Haupteinteilung der Testaufgaben erfolgt in die **gebundene Aufgabenbeantwortung** und die **freie Aufgabenbeantwortung**. Zu den Testaufgaben mit gebundener Aufgabenbeantwortung zählen die *Multiple-Choice-Aufgaben*, die *Zuordnungsaufgaben* so-

wie die *Umordnungsaufgaben*. Aufgabentypen mit freier Aufgabenbeantwortung sind die *Ergänzungsaufgaben* sowie die *Kurzaufsatz-Aufgaben*. Es folgt eine genauere Beschreibung der einzelnen Aufgabentypen. Anschließend wird erläutert, welche Aufgabentypen bei welchen Testarten verwendet werden können.

2.4.1 Multiple-Choice-Aufgaben

Die Multiple-Choice-Aufgabe ist ein häufig verwendeter Aufgabentyp. Dem Testkandidaten werden verschiedene Antwortmöglichkeiten zur Verfügung gestellt, von denen er die Richtige(n) auswählen muß. Die verschiedenen Antwortmöglichkeiten werden auch *Items* genannt. Es werden **Multiple-Choice-Aufgaben mit Mehrfachnennung** und **Multiple-Choice-Aufgaben ohne Mehrfachnennung** unterschieden. Multiple-Choice-Aufgaben mit Mehrfachnennung lassen dem Testkandidaten die Möglichkeit, mehrere richtige Items auszuwählen, wobei die Aufgabe richtig beantwortet wurde, wenn alle richtigen Items ausgewählt wurden (siehe Abbildung 2.1²). Multiple-Choice-Aufgaben ohne Mehrfachnennung lassen lediglich die Auswahl eines Items zu (siehe Abbildung 2.2).

Welche der folgenden Energieträger sind Primärenergieträger?	
<input type="radio"/>	Steinkohle
<input type="radio"/>	Benzin
<input type="radio"/>	Strom
<input type="radio"/>	Erdgas
<input type="radio"/>	Fernwärme
Richtige Lösung: Steinkohle und Erdgas	

Abbildung 2.1: Beispiel einer Multiple-Choice-Aufgabe mit Mehrfachnennung

Welcher Energieträger hatte am Primärenergieverbrauch der BRD von 1990 den größten Anteil?	
<input type="radio"/>	Steinkohle
<input type="radio"/>	Braunkohle
<input type="radio"/>	Mineralöl
<input type="radio"/>	Kernkraft
<input type="radio"/>	Erdgas
Die richtige Lösung ist Mineralöl mit einem Anteil von 35 %	

Abbildung 2.2: Beispiel einer Multiple-Choice-Aufgabe ohne Mehrfachnennung

Ein Sonderfall der Multiple-Choice-Aufgabe ohne Mehrfachnennung ist die **Richtig-Falsch-Aufgabe**, bei der als Antwortmöglichkeiten lediglich “Richtig” oder “Falsch” zur Verfügung stehen. Multiple-Choice-Aufgaben ohne Mehrfachnennung mit lediglich zwei Items werden

²Alle Daten der in den Abbildungen gezeigten Beispiele sind aus [Voß92] entnommen

auch den Richtig-Falsch-Antworten zugeordnet, da eine Umformung der Aufgabe in eine Richtig-Falsch-Aufgabe möglich ist.

2.4.2 Zuordnungsaufgaben

Bei der Zuordnungsaufgabe müssen beide Elemente einer Aufgabe - das Problem und die Lösung - einander zugeordnet werden. Es werden dem Testkandidaten mehrere Fragen und mehrere Antworten zur Verfügung gestellt, die er zu passenden Paaren ordnen muß. Ein einfaches Beispiel sind die Fragen "Zucker ist ...", "Pfeffer ist ..." und "Die Zitrone ist ...", die den Antworten "süß", "sauer" und "scharf" zugeordnet werden müssen. Ein weiteres Beispiel wird in Abbildung 2.3 gezeigt.

Ordnen sie den verschiedenen Verkehrsbereichen ihren Endenergieverbrauch in der BRD von 1990 zu!	
1 Luftverkehr	a 2.066,5 PJ
2 Straßenverkehr	b 89,2 PJ
3 Schienenverkehr	c 195,7 PJ
4 Binnenschifffahrt	d 27,7 PJ
Richtige Lösung: 1c; 2a; 3b; 4d	

Abbildung 2.3: Beispiel einer Zuordnungsaufgabe

2.4.3 Umordnungsaufgaben

Bei der Umordnungsaufgabe muß der Testkandidat Buchstaben, Zahlen, Worte oder Bilder entsprechend einer ursprünglichen Ordnung umordnen. Ein Beispiel ist die Aufgabenstellung "10+3=7", bei der die Zahlen in eine Reihenfolge gebracht werden sollen, in der die Rechenaufgabe wahr wird (also "7+3=10" oder "3+7=10"). Ein weiteres Beispiel wäre die Umordnung vorgegebener Worte zu einen Satz.

Ordnen sie folgende Energieträger nach ihrem Anteil am weltweiten Primärenergieverbrauch von 1989! (Den Energieträger mit dem größten Anteil zuerst)	
- Kohle - Kernenergie - Mineralöl - Gas - Wasserkraft -	
Richtige Lösung: 1. Mineralöl (33%) 2. Kohle (28%) 3. Gas (20%) 4. Wasserkraft (7%) 5. Kernenergie (6%)	

Abbildung 2.4: Beispiel einer Umordnungsaufgabe

2.4.4 Ergänzungsaufgaben

Die Ergänzungsaufgabe läßt eine freie Aufgabenbeantwortung durch die Testkandidaten zu. Es werden keine vorgegebenen Antworten bereitgestellt. Die Beantwortung der Aufgabe erfolgt durch ein Schlüsselwort oder eine Zahl. Ein Beispiel ist die Aufgabe: "Welches ist die Hauptstadt von Frankreich". Als Antwort müssen die Testkandidaten selbstständig "Paris" aufschreiben.

Sonderformen der Ergänzungsaufgabe sind der Lückentext und der C-Test. Beim **Lückentext** müssen fehlende Worte in einen Text eingefügt werden. Der **C-Test** [Rak183] verlangt von den Testkandidaten, in einem verstümmelten Text die fehlenden Worthälften zu ergänzen.

Richtige Lösung: Exergie

Ergänzen sie folgenden Satz:
Der Anteil der Energie, der sich in Arbeit umwandeln läßt, wird als _____ bezeichnet.

Abbildung 2.5: Beispiel einer Ergänzungsaufgabe

2.4.5 Kurzaufsatz-Aufgaben

Bei der Kurzaufsatz-Aufgabe wird die Antwort - schriftlich oder mündlich - von den Testkandidaten frei ausformuliert. Die Kurzaufsatz-Aufgabe ist ein beliebiger Aufgabentyp, allerdings ist die Auswertungsobjektivität solcher Aufgaben kaum gegeben. Es sollte zumindest die Einschränkung auf max. 150 Worte für die Antwort erfolgen [LiRa94], um die Testkandidaten zu einer kurzen und präzisen Antwort zu zwingen.

Erläutern sie den Begriff Primärenergie! (max.50 Worte)

Eine richtige Lösung: Primärenergie ist der Energieinhalt von Energieträgern, die noch keiner Umwandlung oder Behandlung unterworfen wurden.

Abbildung 2.6: Beispiel einer Kurzaufsatz-Aufgabe

2.4.6 Weitere Aufgabentypen

Aufgabentypen der gebundenen Aufgabenbeantwortung und der freien Aufgabenbeantwortung lassen sich kombinieren, in dem die Wahl der gebundenen Aufgabenbeantwortung mit einer freien Aufgabenbeantwortung begründet wird.

Es gibt noch weitere Aufgabentypen, die alle der freien Aufgabenbeantwortung zuzurechnen sind. Dazu zählen Aufgaben, die das Malen eines Bildes verlangen und die Aufgaben eines Materialbearbeitungstests (siehe Abschnitt 2.3). Die rechnerunterstützte Bearbeitung

solcher Aufgaben würden nur schwer zu realisieren sein und nicht mehr das durch diese Aufgaben zu messende Merkmal messen, sondern vielmehr die Fähigkeiten der Testkandidaten, mit dem Rechner umzugehen. Deshalb wird in dieser Arbeit nicht weiter auf diese Aufgabentypen eingegangen.

2.4.7 Zuordnung der Aufgabentypen zu den Testarten

Bei vielen Testarten gibt es keine Einschränkung bezüglich der Aufgabentypen, die in dem Test verwendet werden dürfen. Sie können alle die in Abschnitt 2.4.1 bis Abschnitt 2.4.5 beschriebenen Aufgabentypen enthalten. Zu diesen Testarten zählen der *nichtstandardisierte Test*, der *Leistungs-*, der *Fähigkeits-* und der *Persönlichkeitstest*, der *verbale Test*, der *einfache* und der *komplexe Test*, der *normorientierte* und der *kriteriumsorientierte Test*, der *Eingangs-* und der *Abschlußtest*, der *Test zur Lernfortschrittskontrolle* sowie der *Eignungs-*, der *Schnelligkeits-* und der *Niveautest*.

Standardisierte Tests können prinzipiell auch alle oben erwähnten Aufgabentypen enthalten, wobei standardisierte Tests im Normalfall keine Kurzaufsatz-Aufgaben enthalten, da die Auswertungsobjektivität so nur schwer gewährleistet werden kann. Auch sollten bei standardisierten Tests die Aufgaben lediglich schriftlich gestellt und beantwortet werden, damit die Durchführungsobjektivität gewährleistet ist.

Befragungstests enthalten mündlich gestellte Kurzaufsatz-Aufgaben, die ebenfalls mündlich beantwortet werden.

Paper-Pencil-Tests können alle Aufgabentypen enthalten, die schriftlich gestellt und bearbeitet werden.

Materialbearbeitungstests enthalten die im letzten Teil des Abschnitts 2.4.6 beschriebenen Aufgabentypen und sind damit für diese Arbeit ohne Belang.

Nonverbale Tests können alle Aufgabentypen bis auf die Kurzaufsatz-Aufgabe enthalten, da in diesem Fall eine verbale Beantwortung vorgegeben ist.

Tests mit gebundener Aufgabenbeantwortung können ausschließlich Multiple-Choice-Aufgaben, Zuordnungs- oder Umordnungsaufgaben enthalten.

Tests mit freier Aufgabenbeantwortung können dahingegen lediglich Ergänzungsaufgaben oder Kurzaufsatz-Aufgaben enthalten.

Hochstrukturierte Tests können alle Aufgabentypen eines Tests mit gebundener Aufgabenstellung sowie Ergänzungsaufgaben enthalten.

Niedrigstrukturierte Tests können alle Aufgabentypen enthalten, darunter müssen sich auch Kurzaufsatz-Aufgaben befinden.

Tabelle 2.2 verdeutlicht die Zuordnung zwischen den Aufgabentypen und den Testarten.

2.5 Testdurchführung

Dieser Abschnitt beschäftigt sich mit der Durchführung von Tests. Wie bereits in Abschnitt 2.3 beschrieben, kann die herkömmliche Durchführung eines Tests *mündlich* (Befragungstests), *schriftlich* (Paper-Pencil-Tests) oder *handlungsorientiert* (Materialbearbei-

Testarten	Aufgabentypen				
	Multiple-Choice-Aufgabe	Zuordnungsaufgabe	Umordnungsaufgabe	Ergänzungsaufgabe	Kurzaufsatz-Aufgabe
Standardisierter Test	JA	JA	JA	JA	(JA) ^a
Nichtstandardisierter Test	JA	JA	JA	JA	JA
Leistungstest	JA	JA	JA	JA	JA
Fähigkeitstest	JA	JA	JA	JA	JA
Persönlichkeitstest	JA	JA	JA	JA	JA
Befragungstest	NEIN	NEIN	NEIN	NEIN	JA
Paper-Pencil-Test	JA	JA	JA	JA	JA
Materialbearbeitungstest	NEIN	NEIN	NEIN	NEIN	NEIN
Verbaler Test	JA	JA	JA	JA	JA
Nonverbaler Test	JA	JA	JA	JA	NEIN
Einfacher Test	JA	JA	JA	JA	JA
Komplexer Test	JA	JA	JA	JA	JA
Normorientierter Test	JA	JA	JA	JA	JA
Kriteriumsorientierter Test	JA	JA	JA	JA	JA
Test mit geb. Aufgabenbeantw.	JA	JA	JA	NEIN	NEIN
Test mit ungeb. Aufgabenbeantw.	NEIN	NEIN	NEIN	JA	JA
Hochstrukturierter Test	JA	JA	JA	JA	NEIN
Niedrigstrukturierter Test	JA	JA	JA	JA	JA ^b
Eingangstest	JA	JA	JA	JA	JA
Test zur Lernfortschrittskontr.	JA	JA	JA	JA	JA
Abschlußtest	JA	JA	JA	JA	JA
Eignungstest	JA	JA	JA	JA	JA
Schnelligkeitstest	JA	JA	JA	JA	JA
Niveautest	JA	JA	JA	JA	JA

^aWegen der geringen Auswertungsobjektivität in der Regel nicht verwendet

^bDieser Aufgabentyp muß enthalten sein

Tabelle 2.2: Die Zuordnung der Aufgabentypen zu den Testarten

Ein JA in einem Feld bedeutet, daß der Test der dazugehörigen Zeile den Aufgabentyp der entsprechenden Spalte enthalten kann. Ein NEIN bedeutet, daß der Test den Aufgabentyp nicht enthalten darf.

tungstests) erfolgen.

Es wurde erläutert, dass handlungsorientierte Aufgaben (siehe Abschnitt 2.4.6) nicht rechnerunterstützt bearbeitet werden sollten.

Die rechnerunterstützte Bearbeitung von mündlichen Aufgaben ist heutzutage technisch möglich. Die Atmosphäre einer mündlichen Prüfung, bei der sich Prüferin und Testkandidat gegenüber sitzen, kann jedoch allein durch eine rechnerunterstützte Sprachein- und -ausgabe nicht realisiert werden. Die Durchführung einer A/V-Konferenz zwischen Prüferin und Testkandidat könnte eine geeignete Alternative zu den herkömmlichen Befragungstests sein, übersteigt aber die Intentionen der Aufgabenstellung dieser Arbeit. Die Durchführung von handlungsorientierten und mündlichen Tests wird deshalb nicht erläutert.

Als Gegenstück zum rechnerunterstützten Test bleibt der Paper-Pencil-Test. Die schriftliche Testdurchführung entspricht weitgehend einer rechnerunterstützten Testdurchführung. Die *Durchführung von Paper-Pencil-Tests* ist dadurch gekennzeichnet, daß sich Aufsicht und Testkandidaten zu einem bestimmten Zeitpunkt an einem bestimmten Ort einfinden. In der Regel wird zuerst die Anwesenheit der Teilnehmenden kontrolliert. Anschließend wird die *Testanleitung* bekanntgegeben. Dies kann mündlich oder schriftlich erfolgen. Wichtige Informationen können zusätzlich an Tafeln oder ähnlichem vermerkt werden. Dazu zählt die Dauer des Tests.

Nach der Bekanntgabe der Arbeitsanleitung werden die *Aufgabenblätter* ausgeteilt. Diese können bereits vorher verteilt werden, dürfen aber erst nach der Bekanntgabe der Arbeitsanleitung von den Testkandidaten eingesehen werden.

Es folgt die Bearbeitung der Aufgabenblätter durch die Testkandidaten. Nach der Bearbeitung werden die Aufgabenblätter abgegeben oder von der Aufsicht eingesammelt. Dies muß spätestens nach dem Ende der vorgegebenen Testzeit erfolgen.

Nach der Abgabe wird der Test korrigiert und den Teilnehmenden das Ergebnis des Tests mitgeteilt. Anschließend kann eine Einsichtnahme der Kandidaten in die Testauswertung erfolgen.

2.6 Testauswertung

Die Auswertung eines Tests erfolgt zunächst unabhängig von der Intention des Tests. Dazu werden zuerst die Punkte der einzelnen Testaufgaben ermittelt. Für jede Testteilnehmerin werden die von ihr erzielten Punkte der einzelnen Aufgaben zu einem *Rohwert* zusammengefaßt. Bei der Testauswertung wird nicht von der erreichten Punktzahl sondern von *Rohwerten* gesprochen, weil diese Werte erst später in "aussagefähigere, testspezifische Werte transformiert werden" [Sche80]. Erst nachdem für jeden Teilnehmenden der Rohwert ermittelt worden ist, erfolgt die Bewertung der Ergebnisse abhängig von der Intention des Tests.

Neben der Bewertung der Teilnehmenden kann bei der Auswertung eines Tests auch die Qualität der einzelnen Testaufgaben analysiert werden.

2.6.1 Auswertung einer Aufgabe

Die Auswertung einer Aufgabe unterscheidet sich nach ihrem Aufgabentyp. Zuerst muß geklärt werden, ob die Auswertung der Aufgabe zwei- oder mehrwertig erfolgen soll. In der Regel wird für jede richtig beantwortete Testaufgabe ein Punkt vergeben (*zweiwertige Auswertung*). Besonders komplexe Aufgaben, die eine deutlich längere Bearbeitungszeit verlangen, können bei richtiger Beantwortung auch mehrere Punkte ergeben (*mehrwertige Auswertung*). Nach [LiRa94] sollten nicht mehr als ein Viertel der Aufgaben mit zwei Punkten und ein Achtel der Aufgaben mit drei Punkten verrechnet werden, um Verteilungsirregularitäten zu vermeiden. Lange Aufgaben sollten in Teilaufgaben untergliedert werden, die mit je einem Punkt bewertet werden.

In der Praxis ist es nicht immer wünschenswert, Aufgaben lediglich mit falsch oder richtig zu bewerten. Wurde beispielsweise bei einer Multiple-Choice-Aufgabe mit Mehrfachnennung von zwei richtigen Items lediglich ein (richtiges) Item ausgewählt, so sollte diese Antwort weder mit “Falsch” noch mit “Richtig” gewertet werden. In diesem Fall ist eine mehrwertige Punktverteilung nötig. Dies kann entweder durch die Verteilung von Teilpunkten oder durch die Erhöhung der Punktzahl für jede Aufgabe erfolgen. Bei der Erhöhung der Punktzahl sollte dies für alle Aufgaben erfolgen. Ansonsten werden Aufgaben unberechtigt überbewertet, bei denen die Antwort differenziert bewertet werden muß.

2.6.1.1 Multiple-Choice-Aufgabe ohne Mehrfachnennung

Die einfachste Art der Auswertung findet sich bei Multiple-Choice-Aufgaben ohne Mehrfachnennung. Wurde das richtige Item angekreuzt, wird die Aufgabe als richtig bewertet und die volle Punktzahl vergeben. Wurde ein falsches Item gewählt, wird die Aufgabe als falsch beantwortet bewertet und kein Punkt vergeben. Wurde kein Item gewählt, zählt die Aufgabe als unbearbeitet und es wird ebenfalls kein Punkt vergeben. Die Unterscheidung zwischen falsch beantworteten und unbeantworteten Aufgaben ist für die Eliminierung von zufällig richtig geratenen Antworten bei der Testauswertung von Bedeutung. Wurden mehrere Items ausgewählt, muß die Aufgabe als falsch gelöst gewertet werden. In diesem Fall sollte die Testanleitung kontrolliert und gegebenenfalls verbessert werden.

Ein Nachteil von Multiple-Choice-Aufgaben ohne Mehrfachnennung ist die fehlende Bewertung von Teilwissen. Das Wissen, daß entweder Item zwei oder Item drei richtig ist, aber auf keinen Fall die Items eins, vier und fünf, kann nicht wiedergegeben werden.

Eine Möglichkeit, dieses Teilwissen auszuwerten, ist das *CBAA Dreieck* (Computer-Based Alternative Assessment Dreieck) [PaLeSu94]. Bei diesem Verfahren werden den Teilnehmenden zu jeder Multiple-Choice-Aufgabe ein CBAA Dreieck zu Verfügung gestellt, auf dem sie angeben können, mit welcher Wahrscheinlichkeit ihrer Meinung nach die Items richtig sind (siehe Abbildung 2.7 a)). Abbildung 2.7 b) zeigt die Wahrscheinlichkeitswerte exemplarisch für Item A. Für die Items B und C gelten entsprechend die gleichen Wahrscheinlichkeiten. Ist sich ein Kandidat sicher, daß Item A richtig ist, wählt er die Fläche ganz oben bei A. Dies entspricht dem Ankreuzen von Item A bei einer herkömmlichen Multiple-Choice-Aufgabe. Glaubt er zu wissen, daß Item C falsch ist, kann sich aber nicht

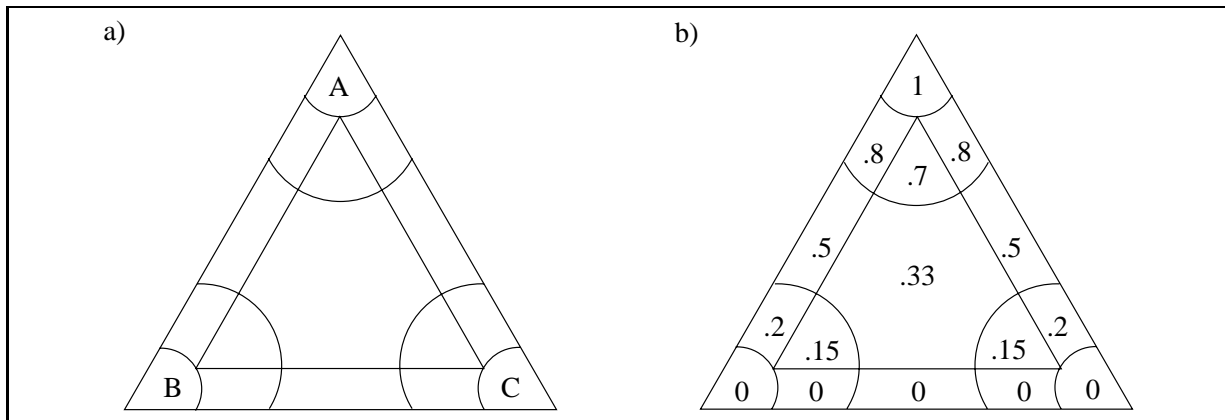


Abbildung 2.7: Computer-Based Alternative Assessment Dreieck

a) zeigt die Abbildung eines CBAA Dreiecks, in b) werden die Wahrscheinlichkeitswerte angegeben, die die einzelnen Bereiche für Item A darstellen

zwischen A und B entscheiden, wählt er die Fläche genau zwischen A und B, hält er bei der Wahl zwischen A und B das Item A für wahrscheinlicher, wählt er die Fläche zwischen A und B, die näher an A liegt, usw.

Ein Nachteil dieses Verfahrens ist die Einschränkung auf drei Items. Mehr Items lassen sich nicht zweidimensional darstellen. Oft werden jedoch zur Reduzierung zufällig richtiger Antworten vier bis fünf Items gefordert (siehe [Sche80], [WeGrHe89]). In diesem Fall könnten die Wahrscheinlichkeitswerte der Items von den Kandidaten "von Hand" als Zahlenwerte angegeben werden. Dazu ist aber ein grundlegendes Verständnis der Teilnehmenden in die Wahrscheinlichkeitsrechnung nötig. Dies kann bei der Mehrzahl der Tests jedoch nicht vorausgesetzt werden. Zudem können auf diese Weise bei der Angabe von falschen Wahrscheinlichkeitswerten Gesamtwahrscheinlichkeiten auftreten, die ungleich eins sind. Eine einfachere Alternative, die nicht die Bandbreite der CBAA Dreiecke umfaßt, ist das Streichen der Items, die die Teilnehmende sicher ausschließen kann, ohne sich damit auf eine bestimmte Antwort festzulegen.

2.6.1.2 Multiple-Choice-Aufgabe mit Mehrfachnennung

Bei einer Multiple-Choice-Aufgabe mit Mehrfachnennung ist die zweiwertige Auswertung (richtig oder falsch) oft unerwünscht. Dieser Aufgabentyp erlaubt eine differenzierte Beantwortung (beispielsweise lediglich eines von drei richtigen Items wählen) und sollte dementsprechend differenziert ausgewertet werden. Dies kann durch die Unterteilung des einen für diese Aufgabe vergebenen Punktes in Teilpunkte oder die Aufstockung der Punktvergabe für jede Aufgabe erfolgen. Es muß genau festgelegt werden, wie teilweise richtige Lösungen gewertet werden. In der Regel wird für jedes falsch ausgewählte Item die Punktzahl eines richtig ausgewählten Items abgezogen. Ansonsten würde die Auswahl aller Items immer die volle Punktzahl ergeben. Die richtig gewählten Items werden im Verhältnis zur Gesamtpunktzahl und der Anzahl der insgesamt richtigen Items gewertet (siehe Formel

2.1).

$$\text{Punktvergabe für ein Item} = \frac{\text{Punktvergabe für die Aufgabe}}{\text{Anzahl richtiger Items}} \quad (2.1)$$

Das Ergebnis einer Aufgabe errechnet sich aus der Summe der richtig gewählten Items minus der Summe der falsch gewählten Items, wobei dieser Wert mit der Punktzahl für ein Item multipliziert werden muß.

Um ungerade Punktverteilungen oder zu kleine Brüche zu vermeiden, kann die Punktvergabe auch gestaffelt werden. Zur Verdeutlichung wird eine Aufgabe mit fünf Items, von denen vier richtig sind, betrachtet. Für die richtig gelöste Aufgabe soll ein Punkt vergeben werden. Für kein oder ein richtig gewähltes Item werden Null Punkte vergeben, zwei bis drei richtige ergeben einen halben Punkt und erst alle vier richtig gewählten Items ergeben den ganzen Punkt. Dadurch wird die Vergabe von Viertelpunkten vermieden.

Wie bei Multiple-Choice-Aufgaben ohne Mehrfachnennung kann bei den Multiple-Choice-Aufgaben mit Mehrfachnennung das Teilwissen der Kandidaten genauer erfaßt werden. Dafür sind keine Konstruktionen wie das CBAA Dreieck nötig, da die Auswahl eines Items unabhängig von der Auswahl der anderen Items erfolgt. Für jedes Item kann neben der Auswahl ebenso das explizite Nichtauswählen erfolgen. Es werden die Auswahlmöglichkeiten “ausgewählt”, “nicht ausgewählt” und “weiß nicht” (keine Auswahl bei diesem Item) für jedes Item zur Verfügung gestellt. Für jedes richtig gewählte Item wird ein (Teil-)Punkt vergeben, auch bei der richtigen Wahl von “nicht ausgewählt”. Jedes falsch gewählte Item, auch die falsche Wahl von “nicht ausgewählt” hat einen (Teil-)Punkt Abzug zur Folge. In diesem Fall ändert sich die Punktvergabe für jedes richtig gewählte Item (siehe Formel 2.2).

$$\text{Punktvergabe für ein Item} = \frac{\text{Punktvergabe für die Aufgabe}}{\text{Anzahl aller Items}} \quad (2.2)$$

Dieses System kann verfeinert werden, indem zwischen “nicht ausgewählt”, “weiß nicht” und “ausgewählt” weitere Zwischenstufen eingeführt werden, bei der die Punktvergabe bzw. der Punktabzug entsprechend abgestuft vorgenommen wird.

2.6.1.3 Zuordnungsaufgaben

Bei der Zuordnungsaufgabe müssen verschiedenen Fragen verschiedene Lösungen zugeordnet werden. Bei der Auswertung kann jede dieser Fragen als Teilaufgabe betrachtet werden und separat mit einem Punkt gewertet werden. Die richtige Zuordnung einer Frage zu einer Lösung ergibt einen Punkt. Wurde die falsche Lösung zugeordnet, wird kein Punkt vergeben und die Frage gilt als falsch beantwortet. Keine Zuweisung ergibt ebenfalls keinen Punkt und die Frage gilt als unbeantwortet. Eine Bewertung des Teilwissens der Kandidaten ist bei Zuordnungsaufgaben ebenfalls möglich. Eine Zuordnungsaufgabe kann in mehrere Multiple-Choice-Aufgaben ohne Mehrfachnennung umgewandelt werden. Da Zuordnungsaufgaben in der Regel mehr als drei verschiedene Lösungen (bei Multiple-Choice-Aufgaben Items) vorgeben, kann das CBAA Dreieck nicht angewandt werden. Das Ausschließen verschiedener Lösungen für eine Frage kann erfolgen.

2.6.1.4 Umordnungsaufgaben

Die Auswertung einer Umordnungsaufgabe gestaltet sich schwierig. Zum einen ist es möglich, daß verschiedene richtige Lösungen existieren. Diese müssen für die Auswertung alle angegeben werden. Zum anderen stellt sich die Frage, wie teilweise richtige Lösungen zu bewerten sind, da bei der Umordnungsaufgabe die Reihenfolge der Elemente entscheidend ist. Ein Beispiel hierfür ist eine Umordnungsaufgabe, bei der verschiedene Länder nach der Anzahl ihrer Einwohner geordnet werden sollen. Wird das Land mit den meisten Einwohnern fälschlicherweise an die letzte Stelle gesetzt, alle anderen aber in der richtigen Reihenfolge geordnet, ist die Aufgabe fast richtig beantwortet. Trotzdem steht keines der Länder an der richtigen Stelle.

Eine Möglichkeit der Auswertung ist, lediglich die vollständig richtigen Lösungen mit einem Punkt zu versehen und sobald eine Abweichung vorliegt Null Punkte zu vergeben. Diese Lösung ist zwar für die Auswertung praktikabel, aber ansonsten keineswegs wünschenswert.

Eine weitere Möglichkeit ist die einmalige Bewertung aller möglichen Alternativen, wobei die Punktvergabe für jede Alternative vorgegeben wird. Allerdings kommen schon bei wenigen Elementen viele verschiedene Lösungs-Alternativen zustande. So existieren bei n Elementen $n!$ Alternativen.

2.6.1.5 Ergänzungsaufgaben

Wurde eine Ergänzungsaufgabe eindeutig gestellt, ist ihre Auswertung einfach. Bereits bei der Aufgabenstellung werden alle richtigen Lösungsmöglichkeiten bestimmt. Auf diese kann bei der Auswertung zurückgegriffen werden. Ein Problem ergibt sich bei Antworten, die bei der Aufgabenstellung nicht bedacht wurden, aber trotzdem nicht als falsch einzuordnen sind. Solche Antworten treten bei schlecht gestellten Aufgaben auf. Hierzu ein Beispiel aus [Sche80]. Die Frage bezieht sich auf die Stadt und lautet: "Martin Luther schlug seine Thesen _____ an." Wird die Aufgabe mit "Kirchentür" ergänzt, ist das nicht falsch. Solche Fragestellungen sind zu vermeiden. Eindeutig könnte die Frage folgendermaßen gestellt werden: "Martin Luther schlug seine Thesen in der Stadt _____ an die Kirchentür an." Diese Aufgabenstellung läßt als einzige richtige Antwort das gesuchte Wittenberg zu.

Lediglich bei der Aufgabenstellung als richtig festgelegte Lösungen können als richtig gewertet werden, wobei verschiedene richtige Lösungen angegeben werden können, die alle die volle Punktzahl ergeben. Auch teilweise richtige Lösungen können angegeben werden, die eine geringere Punktzahl ergeben. Für die Auswertung ist ebenfalls festzulegen, wie Rechtschreibfehler, sowie fehlerhafte Zeit- und Mengenangaben gewertet werden. Infolge der objektiven Auswertung müssen unerwartete (teilweise) richtige Antworten als falsch gewertet werden.

2.6.1.6 Kurzaufsatz-Aufgaben

Die Auswertung einer Kurzaufsatz-Aufgabe ist von allen Aufgabentypen die schwierigste. Eine objektive Auswertung von Kurzaufsatz-Aufgaben ist nicht möglich. Durch das Festlegen eines genauen Antwortschlüssels kann versucht werden, die Auswertung der Aufgabe möglichst objektiv zu gestalten. Aufgrund der frei formulierten Antworten werden aber immer Zweifelsfälle auftreten, bei denen eine Auswertende einen Punkt vergibt, den eine andere bei der gleichen Formulierung nicht mehr geben würde.

Die Auswertung der Aufgabe erfolgt nach dem vorher festgelegten Bewertungsschlüssel. Dort sind die bei der Antwort zu erwähnenden Aspekte und die dazugehörige Punktverteilung aufgelistet. Wurde der Aspekt in der Antwort erwähnt, werden die entsprechenden Punkte vergeben. Die so erreichten Punkte werden zusammengezählt und ergeben die Gesamtpunktzahl der Aufgabe. Zusätzliche Aspekte, die bei der Beantwortung erwähnt wurden, werden bei der Punktvergabe im Regelfall nicht berücksichtigt, auch nicht, wenn sie falsch sind.

2.6.2 Auswertung eines Tests

Die Auswertung eines Tests erfolgt nach der Auswertung der einzelnen Testaufgaben. Im Regelfall besteht die Auswertung eines Tests darin, für jede Teilnehmerin die Punkte, die sie in den einzelnen Aufgaben erhalten hat, zusammenzuzählen und so ihren Rohwert zu ermitteln. Aus den einzelnen Werten wird die *Urliste* ([Sche80]) erstellt, die alle Teilnehmenden und die jeweils erreichten Rohwerte enthält. Diese Urliste stellt die Grundlage für die Bewertung des Tests dar.

Einige Tests messen verschiedene Persönlichkeitsmerkmale, die bei der Auswertung unterschieden werden. Dies kann durch die Zuordnung verschiedener Aufgaben zu verschiedenen Merkmalen oder die Auswertung der gleichen Aufgaben nach verschiedenen Gesichtspunkten erfolgen. In diesem Fall wird nicht ein Rohwert, sondern für jedes Persönlichkeitsmerkmal ein Rohwert ermittelt. Diese Rohwerte müssen getrennt in der Urliste aufgeführt werden.

Um richtig erratene Antworten bei der Bewertung auszuschließen, kann eine *Korrektur des Rohwertes* vorgenommen werden. Ziel dieser Korrektur ist die Gleichstellung der Rohwerte von Kandidaten, die bei Unwissenheit geraten haben, zu den Rohwerten der Kandidaten, die die Aufgaben bei Unwissenheit unbearbeitet gelassen haben. Dies ist bei den Aufgabentypen Ergänzungsaufgabe und Kurzaufsatz-Aufgabe nicht sinnvoll, da ein Erraten der richtigen Lösung wegen der fehlenden vorgegebenen Antwortmöglichkeiten weitgehend ausgeschlossen werden kann. Bei Multiple-Choice-Aufgaben mit Mehrfachnennungen geschieht die Korrektur des Rohwertes bereits, wenn auch negative Punktwerte für einzelne Aufgaben in den Rohwert eingerechnet und nicht mit Null Punkten bewertet werden. Damit werden richtig geratene Items durch falsch geratene Item auch über Aufgabengrenzen hinweg eliminiert. Dieses Mittel sollte lediglich angewandt werden, wenn im Durchschnitt etwa die Hälfte oder mehr der Items eine richtige Lösung darstellen. Ansonsten werden bei Unwissenheit ratende Kandidaten über die Maße benachteiligt.

Um richtig geratene Antworten bei Multiple-Choice-Aufgaben ohne Mehrfachnennung und bei Zuordnungsaufgaben im Verhältnis richtig zu berücksichtigen, gibt es verschiedene Möglichkeiten.

Wie in Abschnitt 2.6.1 bereits erwähnt wurde, werden bei der Aufgabenauswertung richtige und falsche Antworten sowie unbearbeitete Aufgaben unterschieden. Unter der Annahme, daß bei den falsch beantworteten Aufgaben die Lösung geraten wurde, kann davon ausgegangen werden, daß auch bei den richtig beantworteten Aufgaben geratene Lösungen existieren.

Zur Verdeutlichung soll ein Beispiel betrachtet werden. Bei einem Test werden 100 Multiple-Choice-Aufgaben ohne Mehrfachnennung mit je fünf Items gestellt. Jede richtig beantwortete Aufgabe ergibt einen Punkt. Kandidat A und Kandidat B können aufgrund ihres Wissens je 80 Aufgaben richtig beantworten. Bei 20 Aufgaben wissen beide Kandidaten nicht die richtige Lösung. Kandidat A versucht die richtige Lösung bei diesen Aufgaben zu erraten, dagegen läßt Kandidat B diese Aufgaben unbeantwortet. Nach der Wahrscheinlichkeitsrechnung errät Kandidat A bei den 20 Aufgaben viermal die richtige Lösung. Die Wahrscheinlichkeit, die Antwort zu erraten, ist $\frac{1}{\text{Anzahl der Items}}$, in diesem Fall 20% (20 Aufgaben mal 0,2 ergibt vier Aufgaben). Kandidat A erhält bei der Auswertung einen Rohwert von 84 und hat 16 Aufgaben falsch beantwortet. Kandidat B erhält einen Rohwert von 80 und hat 20 unbearbeitete Aufgaben. Ohne eine Korrektur des Rohwerts hat Kandidat A also einen Vorteil.

Die einfachste Möglichkeit der Korrektur des Rohwerts ist, für jede falsch beantwortete Aufgabe eine richtig beantwortete Aufgabe bei der Bewertung außer acht zu lassen. Da eine Aufgabe jedoch im Normalfall mehr als zwei Items enthält, ist die Wahrscheinlichkeit, die richtige Lösung zu raten, kleiner als 50%. Diese Methode würde also die Kandidaten, die bei Unwissenheit die Lösung raten, benachteiligen. In unserem Beispiel hätte Kandidat A nach der Korrektur noch einen Rohwert von 68 (84-16), wohingegen Kandidat B seinen Rohwert von 80 behält.

Statt dessen sollte nicht für jede falsch beantwortete Aufgabe eine richtig beantwortete eliminiert werden, sondern lediglich in einem Verhältnis, das in etwa der durchschnittlichen Anzahl der Items entspricht. Werden in jeder Aufgabe im Durchschnitt fünf Items verwendet, sollte für jede falsche Antwort also lediglich ein Viertel einer richtig beantworteten Aufgabe abgezogen werden. Die allgemeine Formel dazu lautet:

$$\text{Rohwert } X = R_i - \frac{F_i}{m - 1} \quad (2.3)$$

wobei R_i die Anzahl der richtigen Antworten, bzw genauer die Punktvergabe für alle richtigen Antworten, und F_i die Punktvergabe für alle falsch beantworteten Aufgaben darstellt. Die durchschnittliche Anzahl der Items wird durch m symbolisiert. Die Anwendung dieser Formel würde bei unserem Beispiel dazu führen, daß beide Kandidaten einen Rohwert von 80 erhalten. Die Rechnung für Kandidat A lautet: $X_A = 84 - \frac{16}{5-1} = 80$; für Kandidat B: $X_B = 80 - \frac{0}{5-1} = 80$.

Eine weitere Möglichkeit, zufällig richtig geratene Antworten bei der Auswertung im Verhältnis richtig zu berücksichtigen, ist, diese bewußt in den Rohwert zu integrieren. Dazu

wird die erreichbare Punktzahl der unbeantworteten Antworten durch die durchschnittliche Anzahl der Items geteilt und zum Rohwert hinzugezählt. Damit ergibt sich folgende Formel für den Rohwert:

$$\text{Rohwert } X = R_i + \frac{U_i}{m} \quad (2.4)$$

wobei R_i die Punktvergabe für alle richtigen Antworten, U_i die Punktvergabe für alle unbeantworteten Aufgaben und m die durchschnittliche Anzahl der Items pro Testaufgabe darstellt. In unserem Beispiel würde die Anwendung dieser Formel bewirken, daß beide Kandidaten einen Rohwert von 84 erhalten ($X_A = 84 + \frac{0}{5} = 84$; $X_B = 80 + \frac{20}{5} = 84$). Die vorgestellten Methoden und Formeln zur Korrektur des Zufallswertes bei Multiple-Choice-Aufgaben ohne Mehrfachnennung sind entnommen aus [LiRa94].

Ein weiterer Ansatz, mit geratenen Antworten umzugehen, wird bei Wölker in [Wö68] vorgestellt. Für jeden Teilnehmer wird ein *Unredlichkeitsfaktor* ausgerechnet, der das Verhältnis zwischen den richtigen und den falschen Antworten bestimmt. Die Formel lautet:

$$\text{Unredlichkeitsfaktor } U = \frac{\text{falsche gewählte Items}}{\text{richtige gewählte Items}} * \frac{\text{Anzahl richtiger Items}}{\text{Anzahl falscher Items}} \quad (2.5)$$

Wurden keine falschen Items ausgewählt, beträgt der Unredlichkeitsfaktor Null. Je mehr falsche Items im Verhältnis zu den richtigen Antworten gewählt wurden, um so mehr nähert sich der Unredlichkeitsfaktor dem Wert Eins. Werden weniger richtige Items geraten, als aufgrund der Wahrscheinlichkeitsrechnung erwartet werden, kann der Unredlichkeitsfaktor sogar größer als eins werden. Je größer der Unredlichkeitsfaktor ist, um so mehr hat der Teilnehmer geraten. Kandidat B aus unserem Beispiel würde also einen Unredlichkeitsfaktor von Null erhalten, Kandidat A hingegen eine Unredlichkeitsfaktor von 0.05 ($U_A = \frac{16}{84} * \frac{100}{400} = 0.05$).

Bei hohen Unredlichkeitsfaktoren (> 0.25) sollten die entsprechenden Teilnehmenden an einer mündlichen Nachprüfung teilnehmen ([Wö68]). Besteht ein Teilnehmer die Prüfung nicht, muß der vorangegangene Test für diesen Teilnehmer als ungültig erklärt werden, da er lediglich durch Raten die Fragen richtig gelöst hat.

Nachteile dieser Methode sind zum einen der verwaltungstechnische Aufwand, und zum anderen ist bei einer mündlichen Prüfung die Objektivität nicht mehr gewährleistet.

Da die Objektivität ein großer Vorteil von Tests mit Multiple-Choice-Aufgaben darstellt, die bei Einsatz eines Unredlichkeitsfaktors nicht mehr gewährleistet werden kann, sollte diese Methode nicht verwendet werden. Stattdessen sollte entweder eine Korrektur des Rohwertes vorgenommen werden oder der Rohwert ohne Korrekturen verwendet werden, d.h. richtig geratene Antworten werden als richtig gelöste Antworten bewertet. Es gibt verschiedene Untersuchungen zu diesem Thema, die zu unterschiedlichen Ergebnissen kommen (siehe [LiRa94]), so daß hierüber keine eindeutige Aussage gemacht werden kann. Ebenso ist die Frage, ob in der Testanleitung die Teilnehmenden zum Raten aufgefordert oder gewarnt werden sollten, oder ob dieses Thema in der Testanleitung unerwähnt bleiben sollte, von verschiedenen Studien unterschiedlich bewertet worden ([LiRa94]). Meiner Meinung nach sollte aus Gerechtigkeitsgründen eine Korrektur des Rohwertes vorgenommen werden und dies in der Testanleitung angekündigt werden.

2.6.3 Bewertung der Ergebnisse

Die Bewertung der Ergebnisse ist abhängig von dem Zweck des Tests. Dient der Test dem Vergleich des Teilnehmenden mit der durchschnittlichen Leistung einer bestimmten Gruppe (der Norm), erfolgt eine normorientierte Bewertung. Wird der Test dazu verwendet, um zu überprüfen, ob der Testkandidat ein bestimmtes Kriterium erfüllt, erfolgt eine kriteriumsorientierte Bewertung. Bei diesen beiden Bewertungsarten mißt der Test einen *Querschnitt* (zur Zeitachse) der Testkandidaten. Soll die Merkmalsveränderung der Teilnehmenden gemessen werden, muß der *Längsschnitt* gemessen werden, d.h. zu verschiedenen Zeitpunkten müssen Tests mit den gleichen Teilnehmenden durchgeführt werden. Aufgrund dieser Ergebnisse kann eine Bewertung der Merkmalsveränderung erfolgen.

Die Bewertung von Testergebnissen erfolgt nicht gezwungenermaßen an den Teilnehmenden selbst. Es kann durch einen Test ebenfalls eine Lehrveranstaltung, die die Teilnehmenden besuchten, bewertet werden.

2.6.3.1 Normorientierte Bewertung

Die normorientierte Bewertung dient dem Vergleich der einzelnen Teilnehmenden in Bezug auf eine bestimmte Gruppe, der Norm. Bei herkömmlichen Schultests ist dies in der Regel die Klasse oder alle Kinder einer Jahrgangsstufe innerhalb einer Schule. Standardisierte Tests haben eine bedeutend größere Vergleichsgruppe, beim TOFEL-Test (Test of English as a Foreign Language) beispielsweise alle Menschen, die Englisch nicht als Muttersprache gelernt haben. Standardisierte Tests sind normiert, d.h. der Rohwert eines Testkandidaten wird einem Bezugssystem zugeordnet, beispielsweise dem IQ-Wert. Diese Zuordnung entspricht bereits einer Bewertung. Nichtstandardisierten Tests fehlt die Zuordnung der Rohwerte zu einem Bezugssystem. Die Vergleichsgruppe eines nichtstandardisierten Tests bearbeitet den Test in der Regel zum gleichen Zeitpunkt wie der zu bewertende Testkandidat. Die Zuordnung der Rohwerte zu einem Bezugssystem kann also erst nach dem Test durchgeführt werden. Dafür stehen folgende Verfahren zur Verfügung ([GaBe86]):

- Rangordnung,
- Prozentrangplatzierung,
- Standardwert.

Bei der *Rangordnung* wird für jeden Teilnehmenden der *Rang* festgelegt. Der Rang gibt die relative Position des Teilnehmenden innerhalb der getesteten Gruppe an ([Sche80]). Der Teilnehmer mit dem höchsten Rohwert erhält Rang eins, der Teilnehmer mit dem geringsten Rohwert Rang n (bei einer Gruppe von n Personen). Besitzen einige Teilnehmende den gleichen Rohwert, wird der Rang gemittelt. Zur Interpretation eines Rangs muß die Anzahl der Teilnehmenden angegeben werden. Rang zwanzig liegt bei hundert Teilnehmenden im oberen Viertel, bei dreißig Teilnehmenden im unteren Drittel.

Um diese Schwierigkeiten bei der Interpretation zu vermeiden, kann die *Prozentrangplatzierung* angegeben werden. Der *Prozentrang* einer Teilnehmenden berechnet sich aus der

Anzahl der Teilnehmenden, die einen geringeren Rohwert besitzen, dividiert durch die Anzahl aller Teilnehmenden. Dieser Wert mit Hundert multipliziert ergibt den Prozentrang, der unabhängig von der Anzahl der Teilnehmenden interpretiert werden kann.

Ein Nachteil der Prozentrangplatzierung zeigt sich, wenn die Rohwerte der Teilnehmenden eng beieinander liegen. In diesem Fall kann schon eine geringe Veränderung des Rohwertes eine deutliche Änderung des Prozentrangs bewirken. Um dies zu umgehen, kann eine Bewertung aufgrund des *Standardwerts* erfolgen. Der Standardwert einer Teilnehmenden gibt den Abstand ihres Rohwerts vom durchschnittlichen Rohwert an. Die Einheit stellt die *Standardabweichung* dar. Die Formel für den Standardwert lautet ([GaBe86]):

$$\text{Standardwert } z = \frac{\text{Rohwert} - \text{Durchschnittlicher Rohwert}}{\text{Standardabweichung}} \quad (2.6)$$

Zur Bestimmung des durchschnittlichen Rohwerts wird in der Regel das arithmetische Mittel berechnet. Dazu werden alle Rohwerte addiert und diese Summe durch die Anzahl der Rohwerte geteilt. Weitere Verfahren zur Bestimmung der zentralen Tendenz werden in [Sche80] vorgestellt. Die Standardabweichung wird nach folgender Formel berechnet:

$$\text{Standardabweichung } s = \frac{\sqrt{N \sum_{i=1}^N (X_i)^2 - \left(\sum_{i=1}^N X_i\right)^2}}{N} \quad (2.7)$$

In der Formel steht N für die Anzahl der Teilnehmenden und X_i für den Rohwert von Kandidat i (Formel entnommen aus [Sche80]). Der Standardwert bestimmt den relativen Standort der Teilnehmenden und berücksichtigt dabei im Gegensatz zum Prozentrang auch den Abstand zwischen den einzelnen Rohwerten der Teilnehmenden. Standardisierte Tests benutzen prinzipiell das gleiche Verfahren. Der Unterschied liegt darin, daß die Standardabweichung und der durchschnittliche Rohwert bereits festgelegt sind.

Oft wird bei der Bewertung von Testergebnissen eine *Note* vergeben. An deutschen Schulen und Hochschulen ist dabei ein sechstufiges Notensystem üblich, d.h. es können die Noten Eins bis Sechs vergeben werden. Die normorientierte Benotung erfolgt nach einem *Normalverteilungsschlüssel* (vgl. [WeGrHe89]), der festlegt, wieviel Prozent der Schüler welche Note erhalten. In einem Beispiel aus [GaBe86] erhalten 7 % der Teilnehmenden als Note eine Eins, 24 % eine Zwei, 38 % eine Drei, 24 % eine Vier und 7 % eine Fünf. Die Note Sechs wird nicht standardmäßig vergeben. Sie soll lediglich "zur Kennzeichnung extremen Versagens" [WeGrHe89] verwendet werden. In diesem Fall wird die Note Fünf entsprechend weniger oft vergeben. Die Anwendung des Normalverteilungsschlüssels sollte lediglich erfolgen, wenn die Rohwerte eine genügend große Streuung aufweisen. Es ist nicht immer sinnvoll, starr der vorgegebenen Notenverteilung zu folgen. Teilnehmende mit den gleichen Rohwerten sollten immer die gleiche Note erhalten, auch wenn sich der Normalverteilungsschlüssel dadurch nicht mehr einhalten läßt.

2.6.3.2 Kriteriumsorientierte Bewertung

Bei der kriteriumsorientierten Bewertung erfolgt die Bewertung der Teilnehmenden unabhängig voneinander. Es wird für jede Teilnehmende überprüft, ob sie ein bestimmtes Kriterium erfüllt. Ein solches Kriterium ist beispielsweise das Erreichen eines bestimmten Lernziels. Vor der Ausführung des Tests wird bereits festgelegt, welcher Rohwert mindestens erreicht werden muß, um das Kriterium zu erfüllen.

Für die Notengebung bei einer kriteriumsorientierten Bewertung eignet sich besonders ein zweiwertiges Notensystem mit den Noten "bestanden" und "nicht bestanden", wie beispielsweise bei der Führerscheinprüfung. Es kann auch ein mehrwertiges Notensystem verwendet werden, welches statt dem Durchschnitt der Norm das Kriterium als Bezugspunkt für die Benotung verwendet. Das bedeutet, daß schon vor der Testdurchführung ein Notenschlüssel erarbeitet wird, der die Notenvergabe für jeden Rohwert festlegt.

2.6.3.3 Bewertung der Merkmalsveränderung

Die Bewertung der Merkmalsveränderung eines Kandidaten kann erfolgen, wenn mindestens zwei vergleichbare Messungen des gleichen Merkmals vorgenommen werden. In der Regel werden dazu Tests verwendet, die die gleiche Anzahl von Punkten vergeben. Die Merkmalsänderung, in der Regel ein Lernfortschritt, wird durch die Differenz des Rohwerts der ersten und der zweiten Messung ermittelt.

Die Messung der Merkmalsveränderung sollte nicht zu einer Benotung führen, sondern der Motivation der Kandidaten oder der Bewertung von Lehrveranstaltungen dienen. Eine Benotung würde dazu führen, daß alle (unehrlichen) Kandidaten den ersten Test mit Null Punkten abschließen, um so eine möglichst große Merkmalsveränderung vorzugeben. Eine Möglichkeit, dies zu verhindern, liegt in der zusätzlichen Bewertung der einzelnen Tests. Allerdings wird den Kandidaten in der Regel vor der ersten Messung der entsprechende Lehrstoff nicht vermittelt. So erscheint mir eine Bewertung dieses Tests als unangebracht.

2.6.3.4 Bewertung von Lehrveranstaltungen

Zur Bewertung von Lehrveranstaltungen eignen sich besonders Längsschnitt-Messungen. Die Teilnehmenden (oder eine Auswahl der Teilnehmenden) der Lehrveranstaltung müssen bereits vor der Lehrveranstaltung einen Test bearbeiten. Nach (und u.U. auch schon während) der Lehrveranstaltung müssen die gleichen Teilnehmenden einen weiteren Test durchführen, der das gleiche, in der Lehrveranstaltung geschulte Persönlichkeitsmerkmal mißt. Haben sich (fast) alle Teilnehmenden nach dem Besuch der Veranstaltung verbessert, war die Veranstaltung erfolgreich, konnten sich kaum Teilnehmende verbessern, sollte die Qualität der Veranstaltung verbessert werden.

Zur Beurteilung von Lehrveranstaltungen kann auch ein kriteriumsorientierter Test herangezogen werden, den die (oder einige) Teilnehmenden nach der Veranstaltung bearbeiten müssen. Erfüllt die Mehrzahl der Testkandidaten das im Test geforderte Kriterium, welches innerhalb der Lehrveranstaltung vermittelt wurde, war die Lehrveranstaltung erfolgreich.

Kann kaum eine Teilnehmende das Kriterium erfüllen, wurde das Ziel der Lehrveranstaltung nicht erreicht. Diese Art der Beurteilung berücksichtigt nicht das Vorwissen der Teilnehmenden. So kann es sein, daß alle Teilnehmenden schon vor der Veranstaltung das Kriterium erfüllen. In diesem Fall würde selbst die schlechteste Veranstaltung eine gute Bewertung erhalten. Andererseits können Teilnehmende, denen jedes Vorwissen fehlt, trotz einer guten Veranstaltung eine schlechte Bewertung erhalten.

Die Bewertung von Lehrveranstaltungen sollte wegen dieser Abhängigkeit der Bewertung von den Teilnehmenden nicht zur Einstellung schlecht bewerteter Veranstaltungen führen, sondern den Vortragenden zur Einschätzung des Erfolgs ihrer Veranstaltungen dienen und bei schlechtem Abschneiden zu einer Verbesserung der Veranstaltungen führen.

2.6.4 Aufgabenanalyse

Die Aufgabenanalyse dient in erster Linie der Eliminierung oder der Verbesserung ungeeigneter Aufgaben. Bei der Aufgabenanalyse wird der *Schwierigkeitsindex* und die *Trennschärfe* [LiRa94] einer Aufgabe ermittelt.

Der *Schwierigkeitsindex* bestimmt, wieviel Prozent der Teilnehmenden die Aufgabe richtig beantworten konnten und kann für die Einteilung der Aufgaben in verschiedene Schwierigkeitsstufen verwendet werden. Die *Trennschärfe* bestimmt, inwieweit eine Aufgabe zwischen leistungsstarken und leistungsschwachen Teilnehmenden differenziert.

Bei Multiple-Choice-Aufgaben beinhaltet die Aufgabenanalyse zusätzlich die *Distraktorenanalyse* [Sche80], die für die falschen Alternativen der Aufgabe untersucht, ob sie von den leistungsschwachen oder von den leistungsstarken Teilnehmenden gewählt wurden.

2.6.4.1 Schwierigkeitsindex

Der Schwierigkeitsindex berechnet sich aus dem Verhältnis zwischen den Teilnehmenden, die die Aufgabe richtig beantworten konnten und den Teilnehmenden, die die Aufgabe nicht lösen konnten. Die Formel dazu lautet nach [Sche80]:

$$\text{Schwierigkeitsindex } P = 100 * \frac{N_r}{N} \quad (2.8)$$

N steht für die Anzahl aller Teilnehmenden, N_r für die Anzahl der Teilnehmenden, die die Aufgabe richtig beantwortet haben.

Wie bei der Testauswertung kann bei der Aufgabenanalyse von Multiple-Choice-Aufgaben ohne Mehrfachnennung eine Zufallskorrektur vorgenommen werden, um richtig geratene Antworten auszuschließen. In der Regel sind auch bei Unwissenheit der Kandidaten nicht alle Items gleich wahrscheinlich (auch wenn eine gute Aufgabenstellung dies bewirken soll). Dies sollte bei der Zufallskorrektur berücksichtigt werden. Eine entsprechende Formel für die Zufallskorrektur wurde in [Hor71] entwickelt.

$$\text{plausibilitätskorrigierter Schwierigkeitsindex } P_k = 100 * \frac{N_r - N_{f_{max}}}{N} \quad (2.9)$$

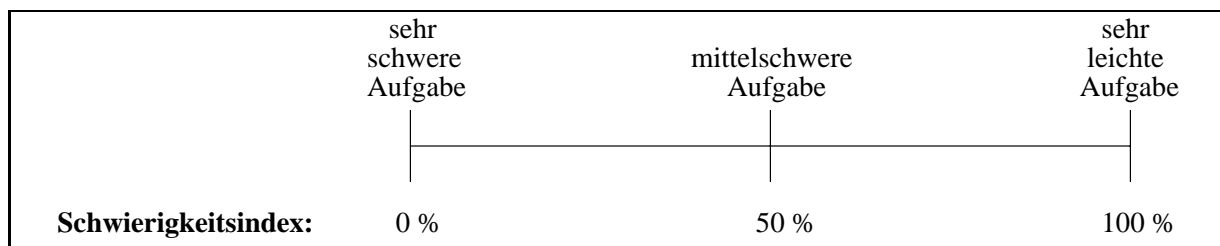


Abbildung 2.8: Schwierigkeitsindex von Aufgaben verschiedener Schwierigkeit

$N_{f_{max}}$ gibt die Anzahl der Teilnehmenden an, die das meistfrequentierte, also das “plausibelste und zugleich falsche” Item gewählt haben. Ist die Wahrscheinlichkeit für alle falschen Items in etwa gleich, kann in Formel 2.9 statt $N_{f_{max}}$ auch $\frac{N_f}{m-1}$ [LiRa94] gewählt werden. N_f steht für die Anzahl der Teilnehmenden, die ein falsches Item gewählt haben, m steht für die Anzahl der Items in der Aufgabe.

Konnte die Aufgabe aufgrund der zu geringen Testzeit nicht von allen Teilnehmenden bearbeitet werden (z.B. bei Schnelligkeitstests), sollte die Bezugsgruppe zur Berechnung des Schwierigkeitsindex nicht die Anzahl aller Teilnehmenden sein, sondern die Anzahl derer, die die Aufgabe bearbeiteten. Dabei gilt die Aufgabe als bearbeitet, falls eine Antwort gegeben wurde oder im Test nach dieser Aufgabe gestellte Aufgaben bearbeitet wurden. Die Formeln ergeben sich aus den Formeln 2.8 und 2.9, indem für N die Anzahl der Teilnehmenden N_b eingesetzt wird, die die Aufgabe bearbeitet haben.

Erfolgt eine mehrwertige Auswertung der Aufgabe, so daß mehrere Punkte oder Teilpunkte vergeben werden, kann der Schwierigkeitsindex folgendermaßen berechnet werden:

$$\text{Schwierigkeitsindex } P = 100 * \frac{\sum_{i=1}^N Pz_i}{N * Pz_{max}} \quad (2.10)$$

Pz_i entspricht der erreichten Punktzahl von Kandidat i in dieser Aufgabe, Pz_{max} der maximal erreichbaren Punktzahl der Aufgabe und N der Anzahl aller Teilnehmenden.

Der Schwierigkeitsindex kann einen Wert zwischen Null und 100 Prozent einnehmen. Ein hoher Schwierigkeitsindex gibt an, daß die Aufgabe leicht zu lösen war. Schwere Aufgaben haben einen niedrigen Schwierigkeitsindex, sie können lediglich von wenigen Teilnehmenden gelöst werden. Abbildung 2.8 verdeutlicht diese Zuordnung. Aufgaben mit einem extrem hohen (100 %) und einem extrem niedrigen Schwierigkeitsindex (0 %) dienen nicht der Differenzierung zwischen den einzelnen Teilnehmenden und sollten in der Regel nicht verwendet werden. Ausnahme sind sogenannte Eisbrecheraufgaben [LiRa94], die am Anfang eines Tests die Angst der Teilnehmenden abbauen sollen und von allen Teilnehmenden richtig gelöst werden können, also einen Schwierigkeitsindex von 100 % erreichen.

2.6.4.2 Trennschärfe

Die Trennschärfe einer Aufgabe gibt an, wie gut eine Aufgabe zwischen leistungsstarken und leistungsschwachen Teilnehmenden eines Tests differenziert und wird über den

Trennschärfekoeffizienten angegeben.

Definition: Der Trennschärfekoeffizient einer Aufgabe ist gleich der Korrelation zwischen dem Aufgabenpunktwert und dem Rohwert eines jeden Teilnehmenden.
(nach [LiRa94])

Dem Trennschärfekoeffizienten liegt folgender Gedanke zugrunde: Jede Aufgabe des Tests kann als eigener Test angesehen werden und mit dem Ergebnis des Tests korreliert werden. Der Trennschärfekoeffizient gibt an, wieweit eine Aufgabe das mißt, was die restlichen Aufgaben zusammen messen.

Zur Berechnung des Trennschärfekoeffizienten kann die Methode der *punktbiserialen Korrelation* (siehe [LiRa94]) verwendet werden. Gilt für die Anzahl der Teilnehmenden N : $N - 1 \approx N$ kann folgende Formel zur Berechnung des Trennschärfekoeffizienten verwendet werden³.

$$pbisr_{jt} = \left(\frac{\sum_{i=1}^{N_r} X_{r_i}}{N_r} - \frac{\sum_{i=1}^N X_i}{N} \right) * \sqrt{\frac{N_r}{N - N_r}} * \frac{N}{\sqrt{N \sum_{i=1}^N X_i^2 - \left(\sum_{i=1}^N X_i \right)^2}} \quad (2.11)$$

Dabei bedeutet in dieser Formel:

X_{r_i} Testrohwert des Teilnehmenden i , der die Aufgabe richtig beantwortet hat,

N_r Anzahl der Teilnehmenden, die die Aufgabe richtig beantwortet haben,

X_i Testrohwert des Teilnehmenden i ,

N Anzahl aller Teilnehmenden. (Formel entnommen aus [LiRa94].)

Können aufgrund einer geringen Bearbeitungszeit nicht alle Teilnehmenden die Aufgabe bearbeiten, sollten als Bezugsgruppe nicht alle Teilnehmenden gewählt werden, sondern lediglich diejenigen, die die Aufgabe bearbeitet haben (wie beim Schwierigkeitsindex). Die Formel ändert sich, indem N (die Anzahl aller Teilnehmenden) durch N_b (die Anzahl der Teilnehmenden, die die Aufgabe bearbeitet haben) ersetzt wird.

Die oben vorgestellten Formeln eignen sich bei Aufgaben, die zweiwertig ausgewertet werden. Bei einer mehrwertigen Auswertung kann keine punktbiserale Korrelation angewandt werden. In diesem Fall muß die Punktvergabe der Aufgabe in eine Rangfolge gebracht werden und mit der Rangfolge der Rohwerte des Tests korreliert werden. Die Formel für den Korrelationskoeffizienten lautet (nach [Sche80]):

$$\text{Korrelationskoeffizient } r = \frac{\sum_{i=1}^N (z_{x_i} * z_{y_i})}{N} \quad (2.12)$$

z_{x_i} und z_{y_i} stehen für die z-Werte der Variablen x_i und y_i . In unserem Fall steht x_i für den Punktwert, den Teilnehmerin i in der Aufgabe erhalten hat, und y_i für den Rohwert der

³bei einer hinreichend großen Teilnehmendenmenge ist dies erfüllt, sonst ist die Berechnung der Trennschärfe wegen des zu kleinen Stichprobenumfangs nicht sinnvoll

Teilnehmerin i . Die z-Werte berechnen sich nach der Formel ([Sche80]):

$$\text{z-Wert der Teilnehmenden } i: z_{x_i} = \frac{X_i - \bar{X}}{s_x} \quad (2.13)$$

X_i ist der Rohwert bzw. Punktwert der Teilnehmenden, \bar{X} der durchschnittliche Roh- bzw. Punktwert und s_x die Standardabweichung. Die Standardabweichung berechnet sich aus Formel 2.7 (siehe Abschnitt 2.6.3.1). Wird die Formel 2.13 in 2.12 eingesetzt, ergibt sich für die Berechnung des Trennschärfekoeffizienten folgende Formel:

$$\text{Trennschärfekoeffizient } r = \frac{\sum_{i=1}^N \left(\left(\frac{Pz_i - \bar{Pz}}{Pz_s} \right) * \left(\frac{X_i - \bar{X}}{X_s} \right) \right)}{N} \quad (2.14)$$

Pz_i ist die von Kandidat i in der Aufgabe erreichte Punktzahl,

\bar{Pz} ist die durchschnittlich erreichte Punktzahl,

Pz_s die Standardabweichung bei der Punktzahl und

X_s die Standardabweichung des Rohwerts.

Der Trennschärfekoeffizient kann einen Wert zwischen -1.0 und 1.0 annehmen. Je höher der Trennschärfekoeffizient ist, um so besser kann die Aufgabe zwischen leistungsschwachen und leistungsstarken Teilnehmenden differenzieren, d.h. um so geeigneter ist die Aufgabe für den Test. Besteht keine Korrelation zwischen den Fähigkeiten der Teilnehmenden und der Aufgabenauswertung (Trennschärfekoeffizient = 0.0), ist die Aufgabe unbrauchbar. Bei einem negativen Trennschärfekoeffizienten könnte die Bewertung der Aufgabe verändert werden. Die richtige Lösung (von den leistungsschwachen Teilnehmenden gewählt) ergibt keinen Punkt und die falsche Lösung (von den leistungsstarken gewählt) ergibt einen Punkt. Damit würde sich ein positiver Trennschärfekoeffizient ergeben. Diese Auswertung ist vor den Teilnehmenden allerdings nicht zu rechtfertigen, und Aufgaben mit einem negativen Trennschärfekoeffizient sollten deshalb ebenfalls nicht verwendet werden.

2.6.4.3 Zusammenhang zwischen Schwierigkeitsindex und Trennschärfekoeffizient

Der Schwierigkeitsindex und der Trennschärfekoeffizient einer Aufgabe sind nicht unabhängig voneinander. Ein Trennschärfekoeffizient von 1.0 kann sich lediglich bei einem Schwierigkeitsgrad von 50 Prozent ergeben. Ein Schwierigkeitsgrad von 50 Prozent hat jedoch nicht einen Trennschärfekoeffizienten von 1.0 zur Folge. Der Schwierigkeitsgrad von 50 Prozent besagt, daß die Aufgabe die Teilnehmenden in zwei gleichgroße Gruppen einteilt. Ein Trennschärfekoeffizient von 1.0 ergibt sich, falls die Hälfte, die die Aufgabe richtig beantwortet hat, lediglich aus den leistungsstarken und die andere aus den leistungsschwachen Teilnehmenden besteht. Enthalten beide Hälften jeweils Teilnehmende, die in der anderen Hälfte zu erwarten wären, sinkt der Trennschärfekoeffizient entsprechend. In Abbildung 2.9 wird dieser Zusammenhang verdeutlicht.

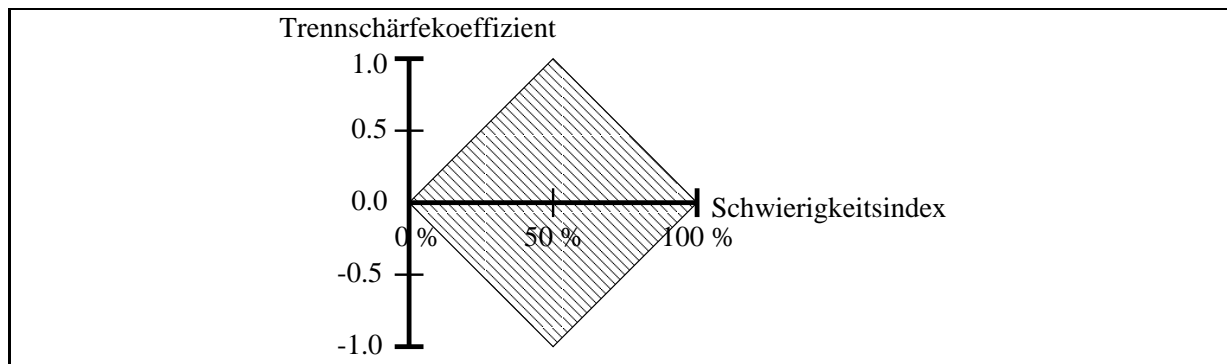


Abbildung 2.9: Der Zusammenhang zwischen dem Schwierigkeitsindex und dem Trennschärfekoeffizient

Der Trennschärfekoeffizient kann abhängig vom Schwierigkeitsindex einen Wert innerhalb der schraffierten Fläche annehmen.

2.6.4.4 Distraktorenanalyse

Die Distraktorenanalyse dient bei Multiple-Choice-Aufgaben der Bewertung der falschen Alternativen, der *Distraktoren*. Von guten Multiple-Choice-Aufgaben wird erwartet, daß ihre Distraktoren in etwa den gleichen Schwierigkeitsindex aufweisen, d.h. gleich oft ausgewählt werden. Wird ein Distraktor fast gar nicht gewählt, ist er offenbar zu leicht als falsche Alternative zu erkennen. Dadurch erhöht sich die Wahrscheinlichkeit, die richtige Alternative zu erraten. Zu attraktive Distraktoren verringern diese Wahrscheinlichkeit.

Alle Distraktoren einer Multiple-Choice-Aufgabe sollten einen negativen oder nahe Null angesiedelten Trennschärfekoeffizienten aufweisen. Ansonsten sind die Alternativen zu schwer zu unterscheiden, so daß auch die leistungsstarken Teilnehmenden sich nicht zwischen der richtigen Alternative und den Distraktoren entscheiden können. Die Bestimmung der Schwierigkeitsindizes und der Trennschärfekoeffizienten erfolgt auf die gleiche Weise wie bei der Aufgabenanalyse.

Distraktoren mit einem zu niedrigen oder zu hohem Schwierigkeitsgrad oder einem positiven Trennschärfekoeffizienten sollten nicht verwendet werden und ggf. nach der Distraktorenanalyse ersetzt werden. Um auf eine weitere Analyse der neu eingesetzten Distraktoren zu verzichten, kann bereits bei der ersten Analyse eine größere Anzahl von Distraktoren eingesetzt werden, als erforderlich ist. Werden die ungeeigneten Distraktoren gestrichen, bleiben genügend Distraktoren für die erneute Testdurchführung übrig.

Kapitel 3

Rechnerunterstützte Tests

In diesem Kapitel wird auf die Besonderheiten des rechnerunterstützten Testens eingegangen. Zuerst wird die Durchführung eines rechnerunterstützten Tests beschrieben. Anschließend werden die Vorteile und die neuen Möglichkeiten ebenso wie die Nachteile und Schwierigkeiten von rechnerunterstützten Tests erläutert. Daraufhin wird die Eignung verschiedener Testaufgaben für rechnerunterstützte Tests untersucht. Zum Abschluß wird auf die Besonderheiten einer datenbankgestützten Testumgebung eingegangen.

3.1 Durchführung von rechnerunterstützten Tests

Die Durchführung von Tests mit dem Rechner kann prinzipiell auf die gleiche Weise erfolgen wie Paper-Pencil-Tests (siehe Abschnitt 2.5). Es werden lediglich die Medien Papier und Stift durch die Medien Bildschirm und Tastatur/Maus ersetzt.

Der Einsatz von verteilten Rechnersystemen bietet darüber hinaus die Möglichkeit, von der herkömmlichen Durchführung abzuweichen. Dies bezieht sich vor allem auf den Ort und die Zeit der Durchführung. Der Ort der Durchführung wird lediglich dadurch eingeschränkt, daß ein Zugang zum verteilten Rechnersystem vorhanden sein muß. Das Problem der fehlenden Aufsicht wird in Abschnitt 3.3.1 erläutert. Die Zeit der Durchführung kann ebenfalls frei gewählt oder vorgegeben werden.

Nachdem sich eine Testkandidatin an einem Rechner angemeldet hat, unterscheidet sich die Durchführung im Vergleich zu einem Paper-Pencil-Test kaum.

Der Zugang zum Test kann über ein Passwort erfolgen, welches die Kandidatin vorher erhält. Möglichkeiten zur Kontrolle, ob tatsächlich die "richtige" Teilnehmerin am Test teilnimmt, werden in Abschnitt 3.3.1 diskutiert.

Nach der Identifizierung/Authentifizierung wird der Kandidatin die Testanleitung schriftlich auf dem Bildschirm angezeigt. Der Bestätigung des Erhalts der Testanleitung folgt die Anzeige der Testaufgaben auf dem Bildschirm. Ab diesem Moment beginnt die Zeitmessung für den Test. Der Kandidatin sollte die verbleibende Zeit stets auf dem Bildschirm angezeigt werden.

Die Bearbeitung endet, sobald die Zeit abgelaufen ist oder die Kandidatin den Test bereits

vorzeitig beendet.

Die Korrektur des Tests kann sofort nach der Bearbeitung rechnerunterstützt erfolgen und die Ergebnisse können der Kandidatin unmittelbar nach der Bearbeitung des Tests (unter Berücksichtigung einer kurzen Berechnungszeit) angezeigt werden. Ergebnis meint in diesem Fall zunächst den Rohwert und keine Note.

Eine mögliche Einsichtnahme der Testkandidaten kann ebenfalls sofort nach der Korrektur des Tests erfolgen. Zusätzlich kann den Kandidaten nach jeder Aufgabe innerhalb des Tests eine Einsichtnahme in die richtige Lösung dieser Aufgabe gestattet werden.

3.2 Vorteile von rechnerunterstützten Tests

Vorteile von rechnerunterstützten Tests zeigen sich zum einen im Vergleich mit herkömmlichen Tests. Zum anderen lassen sich durch rechnerunterstützte Tests neue Testarten realisieren, die mit den herkömmlichen Testverfahren nicht durchzuführen sind.

3.2.1 Vorteile im Vergleich zu Paper-Pencil-Tests

Die Vorteile von rechnerunterstützten Tests im Vergleich zu Paper-Pencil-Tests liegen vor allem bei der Auswertung der Tests:

- Bisher mußten die Tests von Personen ausgewertet werden. Diese Aufgabe wird nun durch den Rechner ausgeführt, wodurch menschliche Arbeitszeit eingespart werden kann.
- Eine fehlerhafte Korrektur kann weitgehend ausgeschlossen werden.
- Durch die Bearbeitung mit dem Rechner werden undeutliche Antworten vermieden. Dies kann bei Paper-Pencil-Tests geschehen. Beispielsweise kreuzt ein Testkandidat bei einer Multiple-Choice-Aufgabe zuerst eine Antwortmöglichkeit an und entscheidet sich dann anders. Dazu streicht er das Kreuz durch und kreuzt eine andere Antwortmöglichkeit an. Wiederholt sich dieser Vorgang, ist es für den Korrektor u.U. nicht mehr nachvollziehbar, welche Antwort letztendlich gewählt wurde. In diesem Fall muß die Aufgabe als nicht beantwortet gewertet werden.
Bei einem rechnerunterstützten Test kann jedoch die Auswahl in einfacher Weise geändert werden.
- Die Ergebnisse des Tests können den Testkandidaten unmittelbar nach der Bearbeitung mitgeteilt werden. Dabei bezeichnet der Begriff "Ergebnis" in diesem Fall nicht unbedingt eine Note, sondern den im Test erreichte Rohwert. Die aus dem Rohwert resultierende Note wird eventuell erst anschließend von einer Prüfungskommission nach der Begutachtung der Ergebnisse aller Teilnehmenden bestimmt.

Neben der Auswertung bietet ein rechnerunterstützter Test auch bei der Bearbeitung selbst Vorteile gegenüber Paper-Pencil-Tests:

- Der Test kann an jedem Ort durchgeführt werden. Prinzipiell ist dies bei einem Paper-Pencil-Test auch möglich. Der Test könnte beispielsweise per Post verschickt und ausgefüllt zurückgesandt werden. Dies erfordert allerdings einen höheren Verwaltungsaufwand. Zudem ist bei dieser Methode nicht einmal die Einhaltung der korrekten Testzeit gewährleistet. Damit ist dieses Verfahren für einen Paper-Pencil-Test nicht praktikabel.
- Fehler der Testkandidaten bei der Bearbeitung können eindeutig korrigiert werden. Beispielsweise kann die Auswahl eines Items einer Multiple-Choice-Aufgabe durch das erneute Anwählen dieses Items zurückgenommen werden.
- Nicht zulässige Antwortmöglichkeiten werden eingeschränkt. Es kann beispielsweise realisiert werden, das bei Multiple-Choice-Aufgaben ohne Mehrfachnennung lediglich eine Antwort gleichzeitig ausgewählt sein kann. Die Auswahl einer weiteren Antwort setzt die vorherige Auswahl zurück. So ein Verfahren ist bei einem Paper-Pencil-Test nicht möglich.
- Ein weiterer Vorteil ist die Materialeinsparung. Es wird kein Papier verbraucht. Der Rechner wird bei der Bearbeitung nicht oder lediglich minimal abgenutzt.

3.2.2 Neue Möglichkeiten durch rechnerunterstützte Tests

Durch den Einsatz von Rechnern zur Durchführung von Tests sind neben den eben genannten Vorteilen auch neue Testaufgaben und sogar neue Testarten möglich.

3.2.2.1 Neue Aufgabentypen

Die durch den Einsatz von Rechnern neu gewonnenen Aufgabentypen zeichnen sich dadurch aus, daß sie ohne den Einsatz von Computer-Technologie nicht oder eingeschränkt durchführbar sind. Dazu zählen die Simulations-Aufgaben, die Reaktions-Aufgaben und die Gedächtnis-Aufgaben.

- **Die Simulations-Aufgabe** wird zur Simulation eines Vorgangs verwendet, bei der der Testkandidat in die Steuerung dieses Vorgangs eingreifen kann. Beispiele hierfür sind die Steuerung eines Flugzeugs oder eines Atomkraftwerkes. Simulations-Aufgaben eignen sich besonders zur Überprüfung, ob der Testkandidat die für ihn vorgesehene Aufgabe in der Realität erfüllen kann. Es können Gefahrensituationen erprobt werden, ohne eine reale Gefährdung zu verursachen.
- **Die Reaktions-Aufgabe** mißt die Reaktionszeit der Testkandidaten, auf Impulse (richtig) zu reagieren. Der Impuls wird durch das Anzeigen eines Bildes, Zeichens oder ähnlichem auf dem Bildschirm oder der Ausgabe eines Tones durch den Lautsprecher des Rechners gegeben. Die Reaktion des Testkandidaten erfolgt, indem der Kandidat ein oder mehrere Zeichen auf der Tastatur betätigt. Welche Reaktionen auf welche Impulse erfolgen

sollen, muß dem Testkandidaten vorher mitgeteilt werden.

Neben Reaktionszeit wird bei der Bewertung auch berücksichtigt, ob die richtige Reaktion erfolgte.

- **Die Gedächtnis-Aufgabe** verlangt von der Teilnehmerin eines Tests, die zuvor angegebenen Zahlen, Buchstaben, Wörter oder ähnliches in der gleichen Reihenfolge zu wiederholen.

Dieser Aufgabentyp muß nicht unbedingt an einem Rechner durchgeführt werden. Die Vorgaben können statt über den Bildschirm auch von einer Aufsicht gegeben werden. Allerdings müßten in diesem Falle alle Teilnehmenden die gleichen Aufgaben zum gleichen Zeitpunkt bearbeiten. Eine Ausführung am Rechner ermöglicht es allen Teilnehmenden, die Reihenfolge und Zeit zur Bearbeitung der einzelnen Aufgaben ihren Fähigkeiten und Begabungen entsprechend anzupassen.

3.2.2.2 Neue Durchführungsmöglichkeiten von Tests

Neben den neuen Aufgabentypen kann durch den Einsatz von Rechnern eine neue Art des Testens erfolgen, wie beispielsweise beim **CAT-Test**. Der Begriff CAT steht für Computerized Adaptive Testing [LuBe95] bzw. Computer-Adaptive Testing [CoHa94]. Die Eigenschaft rechnerunterstützter Tests, die Antworten der Aufgaben sofort auszuwerten, werden hierbei verwendet. Ausgehend von der erzielten Punktzahl bei den vorherigen Aufgaben werden die folgenden Fragen generiert. Waren die Antworten zum größten Teil richtig, folgen schwierigere Fragen. Wurden die meisten Aufgaben falsch beantwortet, wird der Schwierigkeitsgrad der Aufgaben gesenkt.

Durch dieses Verfahren wird für alle Kandidaten ein Test mit einem für sie moderaten Schwierigkeitsgrad möglich. Trotz unterschiedlicher Fähigkeiten der einzelnen Kandidaten werden besonders gute Kandidaten nicht über- und besonders schwache Kandidaten nicht unterfordert. Die Kritik an diesem Verfahren bezieht sich im wesentlichen darauf, daß die Aufgaben von den Kandidaten während des Tests nicht mehr korrigiert werden können. In [LuBe95] wurde dieses Problem untersucht und festgestellt, daß sich durch das Korrigieren der Aufgaben die Noten nicht entscheidend verändern. Gute Kandidaten korrigieren oft richtig, wohingegen schwache Kandidaten auch bei der Korrektur viele Fehler machen. Das Ergebnis wird durch fehlende Korrekturmöglichkeit nicht verfälscht.

Ein weiterer Kritikpunkt ist die zu hohe Bewertung der ersten Aufgaben bei der Einstufung der Schwierigkeitsstufe. Diese Kritik richtet sich aber nicht gegen das Verfahren an sich, sondern an einzelne Realisierungen dieses Verfahrens. Wird nach jeweils n Fragen der Schwierigkeitsgrad neu eingestuft, werden die ersten Aufgaben bei der Einstufung des Schwierigkeitsgrades nicht stärker berücksichtigt als alle anderen.

Erweiterungen des CAT-Verfahrens teilen einen Test in verschiedene Teilgebiete ein, für die der Schwierigkeitsgrad separat bestimmt wird.

Eine weitere Möglichkeit rechnerunterstützter Tests ist die Berücksichtigung der Ergebnisse vorheriger Tests eines Kandidaten bei der Generierung der Testaufgaben für diesen

Kandidaten. Konnte der Kandidat in einem Teilgebiet (fast) alle Fragen richtig beantworten, können ihm beim nächsten Test weniger oder schwierigere Fragen dieses Teilgebiets gestellt werden, da der Kandidat dieses Teilgebiet bereits beherrscht.

3.3 Einschränkungen von rechnerunterstützten Tests

Probleme bei der Durchführung von rechnerunterstützten Tests ergeben sich durch mögliche unredliche Teilnehmende eines Tests. Ferner kann das Programm zur Durchführung eines Tests während der Bearbeitung durch einen Testkandidaten ausfallen.

3.3.1 Die Aufsicht von rechnerunterstützten Tests

Sobald das Ergebnis eines Tests Einfluß auf die Zukunft der Teilnehmenden hat, muß davon ausgegangen werden, daß einige der Teilnehmenden auf unehrliche Weise versuchen, ihre Testergebnisse zu verbessern.

Folglich kann auf eine Aufsicht verzichtet werden, solange die Teilnehmenden eines Tests kein Interesse daran haben, ihre Testergebnisse auf unehrliche Weise zu verbessern. Dies ist bei Tests zur Selbstkontrolle oder zur Bewertung einer Lehrveranstaltung der Fall.

In der Regel müssen die Teilnehmenden jedoch beaufsichtigt werden. Bei herkömmlichen Tests werden die Teilnehmenden während der Testdurchführung von Personen beaufsichtigt. Zusätzlich wird vor oder während der Testdurchführung von der Aufsicht eine Authentifizierung der Teilnehmenden vorgenommen.

Wird bei der Testdurchführung den Teilnehmenden der Ort der Durchführung freigestellt, ist eine Aufsicht der Kandidaten nicht möglich. Schon eine Authentifizierung der Testperson kann nicht gewährleistet werden. Ein Passwort kann unberechtigte Personen davon abhalten, sich unter dem Namen eines Testkandidaten anzumelden. Dies dient lediglich dem Schutz der zu Testenden. Beabsichtigt eine Teilnehmerin einen Betrug, kann sie das Passwort an eine Person weitergeben, die den Test für sie bearbeiten soll. Eine Kontrolle per Kamera würde zum einen die entsprechende Hardware bei den Teilnehmenden erfordern und könnte zum anderen leicht manipuliert werden. Eine Methode zur Manipulation wäre, die Kamera auf die Teilnehmerin (vor einem Rechner sitzend) zu richten, während eine unberechtigte Person an einem anderen Rechner den Test bearbeitet. Sobald von einer Manipulation der Teilnehmenden ausgegangen werden muß, kann die Durchführung des Tests von jedem beliebigen Ort nicht mehr angewandt werden.

Wird der Ort der Durchführung eines Tests eingeschränkt, kann sowohl eine Überprüfung der Testkandidaten als auch eine Aufsicht während der Bearbeitung auf die gleiche Weise wie bei herkömmlichen Tests erfolgen. Wird die Ausführung der Test weiterhin über ein Netzwerk durchgeführt, muß gewährleistet sein, daß sich keine Person außerhalb der Räume als Testperson anmelden kann. Eine Methode hierfür ist, daß die Aufsicht bei der Durchführung ein Passwort eingibt, welches für den Teilnehmenden nicht lesbar ist. Wird

dieses Passwort innerhalb des Netzwerks verschlüsselt versendet, kann die Bearbeitung des Tests durch fremde Personen ausgeschlossen werden.

Der unredliche Einsatz anderer Programme zur Verbesserung des Testergebnisses kann durch die Aufsicht kontrolliert werden, indem sie die Monitore der Teilnehmenden beobachtet. Eine weitere Möglichkeit, die diesen Einsatz ganz ausschließt, ist die Bereitstellung einer Testumgebung, in der lediglich das Testprogramm läuft und kein anderes Programm gestartet werden kann. Diese Alternative ist nicht bei jedem Betriebssystem möglich.

Wird die Bearbeitung des Tests wie oben beschrieben durch Personen beaufsichtigt, kann die gleiche Sicherheit vor Manipulationen wie bei herkömmlichen Tests erreicht werden. Ganz ausgeschlossen werden kann eine Manipulation nie (auch nicht bei herkömmlichen Tests). Bei der Durchführung eines Tests über ein Netzwerk ist darauf zu achten, daß Nachrichten verschlüsselt übertragen werden, um eine Manipulation innerhalb der Übertragung zu verhindern.

3.3.2 Der Ausfall des Testprogramms

Der Ausfall eines Testprogramms kann verschiedene Ursachen haben. Neben einem Programmierfehler können ebenso Fehler der Hardware oder Kommunikationsfehler innerhalb des Netzwerks einen Ausfall hervorrufen. Weiterhin kann die Manipulation eines Testkandidaten (z.B. die Unterbrechung der Stromzufuhr) einen Ausfall bewirken.

Es muß verhindert werden, daß Kandidaten im Falle eines zu erwartenden schlechten Abschneidens den Ausfall des Testprogramms hervorrufen. Ein Szenario, indem dieses Kriterium erfüllt wird, läßt sich wie folgt beschreiben: Die Ergebnisse der beantworteten Aufgaben werden persistent gespeichert und bei einer Weiterbearbeitung des Tests durch den Kandidaten berücksichtigt.

Die durch einen Ausfall hervorgerufene Unterbrechung der Bearbeitung des Tests schränkt jedoch die Durchführungsobjektivität des Tests ein. Deshalb sollte die Ausfallwahrscheinlichkeit und Zeit der Unterbrechung möglichst gering gehalten werden.

3.4 Eignung verschiedener Aufgabentypen für rechnerunterstützte Tests

Für die Durchführung von rechnerunterstützten Tests eignen sich einige Aufgabentypen gut, andere lassen sich dagegen nur mit Schwierigkeiten oder gar nicht integrieren.

Als wichtiges Kriterium zur Eignung eines Aufgabentyps für rechnerunterstützte Tests gilt die objektive Auswertung, die eine rechnerunterstützte Auswertung ermöglicht.

Multiple-Choice-Aufgaben können uneingeschränkt für rechnerunterstützte Tests verwendet werden, da sowohl Multiple-Choice-Aufgaben mit als auch ohne Mehrfachnennung objektiv ausgewertet werden können. Bei Multiple-Choice-Aufgaben ohne Mehrfachnen-

Aufgabentypen	Eignung für rechnerunterstützte Tests		
	gut geeignet	bedingt geeignet ^a	ungeeignet
Multiple-Choice-Aufgaben mit Mehrfachnennung	X		
Multiple-Choice-Aufgaben ohne Mehrfachnennung	X		
Zuordnungsaufgaben	X		
Umordnungsaufgaben ohne die Einbeziehung teilweise richtiger Lösungen bei der Auswertung ^b	X		
Umordnungsaufgaben unter der Einbeziehung teilweise richtiger Lösungen bei der Auswertung		X	
Ergänzungsaufgaben ohne Fehlerkorrektur	X		
Ergänzungsaufgaben mit Fehlerkorrektur	X ^c		
Kurzaufsatz-Aufgaben			X
Simulations-Aufgaben	X		
Reaktions-Aufgaben	X		
Gedächtnis-Aufgaben ohne die Einbeziehung teilweise richtiger Lösungen bei der Auswertung	X		
Gedächtnis-Aufgaben unter der Einbeziehung teilweise richtiger Lösungen bei der Auswertung		X	

^aEine rechnerunterstützte Auswertung ist möglich, erfordert jedoch einen hohen Aufwand bei der Angabe der Auswertungskriterien

^bDer Ausschluß teilweise richtiger Lösungen schränkt die Aussagefähigkeit der Aufgabe u.U. stark ein

^cIn Ausnahmefällen können durch die Korrektur semantisch falsche Lösungsworte als richtig gewertet werden.

Tabelle 3.1: Eignung verschiedener Aufgabentypen für rechnerunterstützte Tests

nung existieren genau $n+1$ Lösungsmöglichkeiten (n ist die Anzahl der Items), mit Mehrfachnennung kommen 2^n verschiedene Lösungsmöglichkeiten zustande, dabei sind viele gleichwertig.

Für die rechnerunterstützte Auswertung genügt die Angabe des bzw. der richtigen Items und bei Mehrfachnennung die Angabe, wieviele (Teil-)Punkte für teilweise richtige Lösungen zu vergeben sind.

Zuordnungsaufgaben eignen sich ebenso wie Multiple-Choice-Aufgaben für rechnerunterstützte Tests, da eine objektive Auswertung möglich ist. Für jede Aufgabe (mit $m \geq n$) existieren genau $\sum_{i=1}^n \left(\frac{n!}{(n-i)!} * \frac{m!}{(m-n+i)!} \right)$ verschiedene Lösungsmöglichkeiten, dabei steht n für die Anzahl der Fragen und m für die Anzahl der zuzuordnenden Lösungen. Die Auswertung erfolgt anhand der zuzuordnenden Paare.

Umordnungsaufgaben können ebenfalls objektiv ausgewertet werden. Es existieren $n!$ verschiedene Lösungsmöglichkeiten bei n verschiedenen Elementen. Falls nur die richtige Lösung bewertet wird, ist die Umordnungsaufgabe gut für rechnerunterstützte Tests geeignet. Sollen jedoch auch teilweise richtige Lösungen bewertet werden, muß für jede Lösungsmöglichkeit die Punktzahl angegeben werden. Bei $n!$ Lösungsmöglichkeiten wird dies bereits für kleine n (ab $n \geq 5$) sehr aufwendig und ist damit lediglich bedingt für eine objektive Auswertung und den Einsatz von rechnerunterstützten Test geeignet.

Ergänzungsaufgaben eignen sich gut für rechnerunterstützte Tests, falls als Lösung einzig das richtig geschriebene Lösungswort akzeptiert wird. In diesem Fall ist eine objektive Auswertung einfach zu realisieren.

Im Regelfall sollten die Rechtschreibung und Tippfehler der Teilnehmenden nicht (mit-) gemessen werden, so daß ein falsch geschriebenes Lösungswort als korrekte Lösung bewertet wird. Wird die akzeptierbare Abweichung vom Lösungswort durch Tipp- und Rechtschreibfehler von einem Algorithmus bestimmt, eignen sich Ergänzungsaufgaben auch unter Einbeziehung falsch geschriebener Lösungsworte gut für rechnerunterstützte Tests.

Kurzaufsatz-Aufgaben Kurzaufsatz-Aufgaben können nicht objektiv ausgewertet werden (siehe Abschnitt 2.6.1.6) und sind deshalb nicht für rechnergestützte Tests geeignet.

Simulationsaufgaben sind erst durch den rechnerunterstützten Einsatz möglich. Ein Fehlverhalten kann leicht ausgewertet werden (z.B. Absturz eines Flugzeugs), die Auswertung einer teilweise richtigen Lösung (z.B. eine nicht perfekte Landung) ist dagegen schwieriger. Simulationsaufgaben stellen eine spezielle Form des Testens dar und werden in der Regel nicht zusammen mit traditionellen Testarten verwendet.

Reaktions-Aufgaben können objektiv ausgewertet werden, wobei einerseits die Zeit gemessen wird, die bis zur Reaktion erfolgte und andererseits die Art der Redaktion. Bei der

Auswertung wird für jede Aufgabe ein Wertepaar angegeben, das bei der Bewertung entsprechend gedeutet werden muß. Reaktionsaufgaben eignen sich gut für rechnerunterstützte Tests, zur Messung der Reaktionszeit ist eine rechnerunterstützte Bearbeitung Voraussetzung.

Gedächtnis-Aufgaben können ohne Probleme objektiv ausgewertet werden, falls die Bearbeitung der Aufgabe bei einer falschen Auswahl abgebrochen wird. In diesem Fall wird die Anzahl der richtig gewählten Elemente gemessen. Diese Ausprägung der Gedächtnis-Aufgabe eignet sich gut für rechnerunterstützte Tests.

Wird die Bearbeitung bei einer falschen Wahl nicht abgebrochen, ergeben sich m^n verschiedene Lösungsmöglichkeiten, wobei m die Anzahl aller möglichen Elemente und n die Anzahl der zu wiederholenden Elemente ist. Für die Bewertung von teilweise richtigen Lösungen muß für jede dieser Lösungsmöglichkeiten die entsprechende Punktvergabe angegeben werden (wie bei der Umordnungsaufgabe). Gedächtnis-Aufgaben, die bei einer Fehlwahl nicht abbrechen, sind damit für rechnerunterstützte Tests bedingt geeignet.

3.5 Datenbankgestützte Testumgebungen

Das Abspeichern der Aufgaben, der Tests sowie der Ergebnisse der einzelnen Teilnehmenden in einer Datenbank bietet Vorteile im Vergleich zur Speicherung in einzelnen Dateien. Diese Vorteile zeigen sich sowohl bei der Erstellung und Durchführung eines Tests als auch bei der Auswertung der Ergebnisse.

3.5.1 Erstellung eines Tests

Wenn die Aufgaben verschiedener Tests in einer Datenbank gespeichert werden, ergibt sich eine Aufgabensammlung, auf die mit einer standardisierten Zugriffssprache (SQL) zugegriffen werden kann. Dadurch können Aufgaben bei der Erstellung neuer Tests leicht wiederverwendet werden.

3.5.2 Durchführung von Tests

Das Abspeichern der Ergebnisse einzelner Aufgaben in einer Datenbank garantiert eine persistente Speicherung. Auf diese Weise kann das Ergebnis jeder bearbeiteten Aufgabe sofort gespeichert werden und alle gespeicherten Ergebnisse können nach einem Ausfall des Testprogramms weiterverwendet werden. Dieser wichtige Aspekt des möglichen Datenverlusts bei einem Programmausfall muß bei der Implementierung einer datenbankgestützten Testumgebung nicht weiter berücksichtigt werden. Es muß bei einem Neustart des Programms lediglich kontrolliert werden, ob bereits Ergebnisse gespeichert wurden und diese Ergebnisse müssen ggf. weiterverwendet werden.

Ein weiterer Vorteil bei der Durchführung ergibt sich aus dem Zugriff auf vorangegangene Ergebnisse der Teilnehmenden durch das Datenbanksystem. So kann bei einer ausrei-

chenden Auswahl an Aufgaben verhindert werden, daß bereits bekannte Aufgaben gestellt werden. Ebenso ist es möglich, im Rahmen von Tests zur Prüfungsvorbereitung vermehrt vorher falsch beantwortete Aufgaben zu verwenden. Dies ist prinzipiell auch bei der Abspeicherung der Ergebnisse in Dateien möglich¹, allerdings muß in diesem Fall die Verwaltung der Dateien durch das Testprogramm erfolgen.

3.5.3 Auswertung der Tests

Neben der Ermittlung des Rohwerts für die einzelnen Kandidaten können in einer datenbankgestützten Testumgebung mittels der Datenbank-Anfragesprache weitere Statistiken erstellt werden. Durch die entsprechenden SQL-Anfragen können die einzelnen Aufgaben analysiert werden. Ein Datenbankschema ermöglicht, daß auch Personen, die mit dem Testprogramm und der genauen Speicherstruktur nicht vertraut sind, mit geringem Aufwand ad-hoc eigene Analysen der Ergebnisse vornehmen können.

¹Genaugenommen speichert ein Datenbanksystem die Daten auch in Dateien, die Verwaltung der Dateien wird aber durch das Datenbanksystem übernommen und muß nicht erneut implementiert werden. Insbesondere verwenden Datenbanksysteme verschiedene Techniken, die auch bei einem Systemausfall während des Schreibens der Daten ein Verlust von bereits geschriebenen Daten verhindert. Zusätzlich ermöglichen Datenbanksysteme ein effizientes Suchen und kontrollierte Lese/Schreibzugriffe mit der Möglichkeit der Konsistenzerhaltung. Eine Beschreibung der in Datenbanksystemen verwendeten Konzepte und Techniken findet sich in [GrRe93].

Kapitel 4

Anforderungsanalyse

In diesem Kapitel werden die zu unterstützenden Testarten und Aufgabentypen beschrieben. Anschließend werden die Anforderungen an die Werkzeuge dargestellt. Dabei werden ein Werkzeug zur Aufgaben- und Testadministration, ein weiteres zur Testdurchführung sowie ein Werkzeug zur Auswertung von Tests unterschieden.

4.1 Testarten

Gemäß der Aufgabenstellung wird der *CAT-Test* unterstützt, der eine Generierung von Testfragen erfordert. Es werden verschiedene Ausprägungen unterstützt, die bei der Testerstellung spezifiziert werden müssen. Dazu zählen die Anzahl der erwünschten Schwierigkeitsstufen sowie die Kriterien zum Wechseln der Schwierigkeitsstufe.

Das Generieren der Aufgaben kann unabhängig von den Ergebnissen der vorherigen Aufgaben erfolgen. Diese Durchführungsart, im folgenden *Test mit Aufgabengenerierung* genannt, muß ebenfalls unterstützt werden. Die Generierung der Aufgaben kann in diesem Fall für jede Teilnehmende einzeln oder für alle Teilnehmenden eines Testdurchlaufs gemeinsam erfolgen.

Werden bei einem Test mit Aufgabengenerierung für alle Teilnehmenden die gleichen Aufgaben generiert, ergibt sich ein "klassischer" Paper-Pencil-Test, der in diesem Fall statt mit Papier und Stift rechnerunterstützt durchgeführt wird. Analog können alle in Abschnitt 2.3 vorgestellten Testarten eingesetzt werden, sofern sich durch die unterstützten Aufgabentypen keine Einschränkungen ergeben.

4.2 Aufgabentypen

Wie bereits in der Aufgabenstellung gefordert, sollen die Aufgabentypen Multiple-Choice-Aufgabe und Lückentext unterstützt werden. Werden die Multiple-Choice-Aufgaben genauer betrachtet und die allgemeinere Form des Lückentextes - die Ergänzungsaufgabe - gewählt, ergeben sich folgende zu unterstützende Aufgabentypen:

- Multiple-Choice-Aufgabe ohne Mehrfachnennung,
- Multiple-Choice-Aufgabe mit Mehrfachnennung,
- Ergänzungsaufgabe.

Die Entscheidung, keine weiteren Aufgabentypen zu unterstützen, wird im folgenden anhand der Aufgabentypen näher erläutert.

Zuordnungsaufgaben werden nicht unterstützt, da sie in Multiple-Choice-Aufgaben umgewandelt werden können. Der Vorteil einer Zuordnungsaufgabe gegenüber einer Multiple-Choice-Aufgabe liegt in der Platzersparnis bei der Aufgabenstellung. Dies ist bei einem rechnerunterstützten Test nicht von Belang.

Umordnungsaufgaben werden ebenso wie Kurzaufsatz-Aufgaben bei einem rechnerunterstützten Test nicht verwendet (siehe Abschnitt 3.4), da eine objektive Auswertung nur schwer durchführbar ist.

4.3 Werkzeug zur Test- und Aufgabenadministration

Die Anforderungen an das Werkzeug zur Test- und Aufgabenadministration gliedern sich in die beiden Bereiche Aufgabeneditor und Testeditor.

4.3.1 Aufgabeneditor

Der Aufgabeneditor dient zur Eingabe neuer Testaufgaben und zum editieren bereits bestehender Aufgaben.

4.3.1.1 Eingabe neuer Aufgaben

Bei der Eingabe von Aufgaben muß zunächst der Typ der Aufgabe angegeben werden. Es kann aus den in Abschnitt 4.2 beschriebenen Aufgabentypen gewählt werden. Neben dem Aufgabentyp können für alle Aufgaben die Schwierigkeitsstufe und die Aufgabengebiete gewählt sowie die Punktzahl¹ bestimmt werden. Grundsätzlich muß die Möglichkeit zur Speicherung der Aufgabe sowie zum Abbruch der Eingabe vorhanden sein. Eine Speicherung darf erst erfolgen, wenn alle erforderlichen Angaben gemacht wurden.

Multiple-Choice-Aufgaben ohne Mehrfachnennung Bei diesem Aufgabentyp müssen die Fragestellung sowie die verschiedenen Items spezifiziert werden. Die Anzahl der Items kann frei gewählt werden, muß aber ≥ 2 sein. Es ist erforderlich anzugeben welches Item die richtige Lösung darstellt, und ob die Reihenfolge der Items verändert werden darf².

¹In der Regel wird ein Punkt pro Aufgabe vergeben. Bei Ergänzungsaufgaben ist es sinnvoll, für jede Lücke einen Punkt zu vergeben

²Erscheinen den Kandidaten zwei Items gleichwahrscheinlich, wird in der Regel das erste Item gewählt [Ehe77]. Deshalb sollte in diesem Fall die Reihenfolge der Items festgelegt werden, um die Beantwortung

Multiple-Choice-Aufgaben mit Mehrfachnennung Nach der Wahl dieses Aufgabentyps müssen die Fragestellung sowie verschiedener Items (mindestens eins, ansonsten beliebig viele) eingegeben werden. Zusätzlich ist es erforderlich, für jedes Item die Korrektheit zu bestimmen und anzugeben, ob die Reihenfolge der Items verändert werden darf.

Ergänzungsaufgabe Wurde dieser Aufgabentyp gewählt, muß ein Text eingegeben werden, der mehrere Ergänzungen (“Lücken”) enthält. Für jede Lücke werden die Lösungsworte spezifiziert und für jedes Lösungswort bestimmt, ob eine Korrektur der Rechtschreibung vorgenommen werden soll.

4.3.1.2 Ändern der Aufgaben

Vor der Modifikation der Aufgabe muß über das Werkzeugs die entsprechende, bereits eingegebene Aufgabe ausgewählt werden. Neben der Aufgabe selbst sollten ebenfalls Statistiken über die Aufgabe angezeigt werden können, die von dem Werkzeug zur Auswertung von Tests erstellt werden. Alle Teile der Aufgabe können abhängig vom Aufgabentyp verändert werden. Ebenfalls sollten die Veränderungen an der Aufgabe zurückgenommen und die Bearbeitung ohne zu speichern beendet werden können.

4.3.2 Testeditor

Analog zum Aufgabeneditor kann der Testeditor sowohl zum Eingeben eines neuen Tests als auch zur Veränderung eines bereits erstellten Tests verwendet werden.

4.3.2.1 Eingabe eines neuen Tests

Bei der Testerstellung wird zuerst die Testart festgelegt. Es kann zwischen einem CAT-Test und einem Test mit Aufgabengenerierung gewählt werden. Die weiteren Anforderungen für die Testeingabe unterscheiden sich teilweise zwischen den beiden Testarten, die Eingabe der Auswertungskriterien und der Testanleitung sowie die Auswahl der Teilnehmenden ist in beiden Fällen gleich.

CAT-Test Zur Testerstellung müssen zuerst die Aufgabengebiete des CAT-Tests gewählt und die Anzahl der Schwierigkeitsstufen für jedes Aufgabengebiet bestimmt werden. Zusätzlich wird spezifiziert, wieviele Aufgaben insgesamt pro Aufgabengebiet von den Teilnehmenden bearbeitet werden sollen³ und nach wieviel Aufgaben und welchen Kriterien der Schwierigkeitsgrad gewechselt wird. Weiter muß die Gewichtung der verschiedenen Schwierigkeitsstufen für die Auswertung angegeben werden.

der Frage unabhängig von der Generierung zu machen.

³Die Anzahl der Aufgaben wird indirekt über die Anzahl der erreichbaren Punkte beschrieben. Falls bei der Generierung Aufgaben gewählt werden, die mehrere Punkte ergeben, sinkt die Anzahl der Aufgaben.

Anschließend kann in beliebiger Reihenfolge die Auswahl der Teilnehmenden, der Testaufgaben, sowie die der Kriterien für die Auswertung und die Eingabe der Testanleitung erfolgen.

Test mit Aufgabengenerierung Bei einem Test mit Aufgabengenerierung werden zuerst die verschiedenen Aufgabengebiete gewählt und anschließend für jedes Aufgabengebiet über die Anzahl der erreichbaren Punkte bestimmt, wieviel Aufgaben welcher Schwierigkeitsstufe bearbeitet werden sollen. Zusätzlich wird bestimmt, ob die Generierung der Aufgaben für jeden Kandidaten einzeln erfolgen soll oder die Aufgaben für alle Teilnehmenden eines Testdurchlaufs gleich bleiben sollen, also pro Testdurchlauf einmal generiert werden. Es wird bestimmt, ob die Aufgaben sofort bearbeitet werden müssen oder ein Wechseln zwischen den Aufgaben möglich ist⁴.

Wie beim CAT-Test können anschließend die Teilnehmenden, die Aufgaben sowie die Auswertungskriterien in beliebiger Reihenfolge ausgewählt werden und die Eingabe der Testanleitung erfolgen.

Auswahl der Teilnehmenden Die Auswahl der Teilnehmenden eines Tests erfolgt jeweils für eine *Instanz* dieses Tests. Eine Instanz wird über zwei Datumsangaben bestimmt, die angeben, ab wann diese Instanz des Tests bearbeitet werden darf und bis wann die Bearbeitung erfolgen muß. Zusätzlich wird bestimmt, ob eine Aufsicht beim Test erforderlich ist. In diesem Fall muß ein Passwort für die Aufsicht bestimmt werden. Weiter wird für eine Testinstanz bestimmt, ob es sich um einen bewerteten Test oder eine Übung handelt. Im ersten Fall wird bei der Aufgabengenerierung versucht, für jeden Teilnehmenden von ihm noch nicht bearbeitete Aufgaben zu wählen und, falls dies nicht möglich ist, wird unabhängig von den Lösungen aus den bereits bearbeiteten Aufgaben gewählt. Falls der Test lediglich als Übung gilt, werden für die Aufgabengenerierung für jeden Kandidaten bevorzugt von ihm falsch bearbeitete Aufgaben gewählt, wohingegen öfters richtig bearbeitete Aufgaben kaum verwendet werden.

Die Teilnehmenden einer Instanz des Tests können aus allen bisher eingegebenen Teilnehmenden gewählt werden und zusätzlich können neue Teilnehmende ergänzt werden. Zur Erleichterung der Auswahl der Teilnehmenden sollte es möglich sein, die Teilnehmenden nach bestimmten Kriterien (z.B. alle Personen, die einen bestimmten Kurs besuchen) zu bestimmen.

Auswahl der Testaufgaben Die Aufgaben eines Tests werden aufgrund ihres Aufgabengebiets bestimmt. Bei der Testerstellung muß es möglich sein, neue Aufgabengebiete zu erstellen und bereits vorhandene Aufgabengebiete oder einzelne Aufgaben diesem Aufgabengebiet zuzuordnen. Zusätzlich sollten neue Aufgaben (über den Aufgabeneditor) eingegeben werden können und dem Aufgabengebiet zugeordnet werden können.

⁴Da die Ergebnisse vorheriger Aufgaben hier für die Aufgabengenerierung nicht von Belang sind, können die Aufgaben in beliebiger Reihenfolge bearbeitet werden.

Für jede Schwierigkeitsstufe dieses Aufgabengebiets sollte angezeigt werden, wieviel Aufgaben mit welcher Punktzahl bereits existieren. Eine Abspeicherung des Tests darf erst erfolgen, wenn bei einem *CAT-Test* von jeder berücksichtigten Schwierigkeitsstufe eines Aufgabengebiets die Anzahl der insgesamt in diesem Aufgabengebiet zu erreichenden Punkte mit den vorhandenen Aufgaben generiert werden kann. In einem *Test mit Aufgabengenerierung* muß für jede Schwierigkeitsstufe in jedem Aufgabengebiet genau die erreichbare Punktzahl mit den vorhandenen Aufgaben generierbar sein.

Eingabe der Auswertungskriterien Für die Auswertung eines Tests muß angegeben werden, ob für teilweise richtige Lösungen Teilpunkte vergeben werden.

Ferner wird angegeben, ob für Multiple-Choice-Aufgaben ohne Mehrfachnennung eine Korrektur des Rohwerts vorgenommen werden soll, und ggf. wird das entsprechende Verfahren bestimmt (Abzug eines Punktes oder nach Formel 2.3 oder Formel 2.4 aus Abschnitt 2.6.2).

Für Multiple-Choice-Aufgaben mit Mehrfachnennung muß angegeben werden, ob negative Punkte über die Aufgabengrenze hinweg vergeben werden.

Eine Bewertung der Ergebnisse (Notenvergabe) kann durch das Werkzeug zur Auswertung eines Tests nach der Durchführung des Tests erfolgen.

Eingabe der Testanleitung Für die Testanleitung kann ein Text eingegeben oder eine bereits existierende Testanleitung aus einer Liste ausgewählt werden.

Bestimmung der Testdauer Es ist nicht vorgesehen, für jede Aufgabe die erwartete Bearbeitungszeit anzugeben, da die Bearbeitungszeit indirekt über die zu erreichende Punktzahl bestimmt wird. Folglich muß für einen Test die Bearbeitungszeit angegeben werden.

4.3.2.2 Ändern eines Tests

Für die Änderung eines Tests wird zuerst der entsprechende Test ausgewählt. Ein bereits durchgeführter Test darf nicht mehr verändert werden, da sonst die Testergebnisse Teilnehmenden, die den Test bereits bearbeitet hatten, mit dem der neuen Teilnehmenden nicht mehr miteinander verglichen werden können. Soll ein bereits durchgeführter Test modifiziert werden, muß er nach der Veränderung als neuer Test gespeichert werden.

Nach der Auswahl des Tests sollen alle Eingaben der Testerstellung (bis auf die Wahl der Testart) verändert werden können. Ein Abspeichern der Veränderungen darf erst erfolgen, wenn alle Daten konsistent sind. Wurde beispielsweise die Anzahl der Aufgaben erhöht, darf erst abgespeichert werden, wenn für alle Schwierigkeitsstufen die erforderliche Anzahl der Aufgaben eingegeben wurde.

Für jeden Test können neue Testinstanzen, also neue Durchgänge eines Tests, erstellt werden. Dazu muß angegeben werden, welche Teilnehmenden an der Testinstanz teilnehmen sollen und von wann bis wann die Durchführung erfolgen sollte. Zusätzlich wird bestimmt, ob eine Aufsicht erforderlich ist.

4.4 Werkzeug zur Testdurchführung

Das Werkzeug zur Testdurchführung muß von einem Java-fähigen World Wide Web - Browser ausführbar sein.

Vor der Durchführung eines Tests muß sich die Teilnehmende durch ihren Namen und ein Passwort authentifizieren. Im Falle eines überwachten Tests muß die Aufsicht ebenfalls ein Passwort eingeben, bevor die Testdurchführung gestartet werden kann.

Nachdem sich die Teilnehmende angemeldet hat, wird ihr zuerst die Testanleitung angezeigt, die sie bestätigen muß. Anschließend beginnt die Bearbeitung. Während der Bearbeitung wird der Teilnehmenden die verbleibende Zeit sowie die Anzahl der bearbeiteten und der verbleibenden Aufgaben angezeigt. Daneben wird die aktuell zu bearbeitende Aufgabe angezeigt. Die Bearbeitung erfolgt per Maus (Multiple-Choice-Aufgaben) oder Tastatur (Ergänzungsaufgaben). Die Bearbeitung der Aufgabe endet mit der Bestätigung durch die Teilnehmerin, sie kann auch unbeantwortet beendet werden. Falls bei der Testerstellung angegeben, kann zwischen den Aufgaben gewechselt werden, d.h. die Reihenfolge der Bearbeitung kann selbst bestimmt werden. Die Bearbeitung des Tests endet, sobald die Bearbeitungszeit abgelaufen ist oder die letzte Aufgabe bearbeitet wurde. Der Teilnehmerin wird eine entsprechende Meldung und nach deren Bestätigung der erreichte Rohwert angezeigt.

Die Aufgabengenerierung erfolgt gemäß den bei der Testerstellung angegebenen Kriterien.

4.5 Werkzeug zur Auswertung

Das Werkzeug zur Auswertung muß sowohl die einzelnen Aufgaben als auch den gesamten Test auswerten und analysieren.

4.5.1 Aufgabenauswertung

Die Aufgabenauswertung unterscheidet sich nach dem Aufgabentyp:

4.5.1.1 Multiple-Choice-Aufgaben ohne Mehrfachnennung

Wurde das richtige Item ausgewählt, muß die Aufgabe mit "richtig" bewertet werden, keine Auswahl führt zur Bewertung "nicht bearbeitet" und ein falsch gewähltes Item zur Bewertung "falsch".

4.5.1.2 Multiple-Choice-Aufgaben mit Mehrfachnennung

Werden keine Teilpunkte vergeben, führt die Auswahl aller richtigen Items zur Bewertung "richtig", alle anderen Lösungsmöglichkeiten zur Bewertung "falsch".

Werden Teilpunkte vergeben, ergibt sich die Punktzahl aus der Formel:

$$\text{Punktvergabe} = (\text{richtig gewählte Items} - \text{falsch gewählte Items}) * \frac{\text{erreichbare Punkte}}{\text{Anzahl richtiger Items}} \quad (4.1)$$

Durch diese Formel können negative Teilpunkte entstehen, die je nach Testauswertungsart für die Rohwertermittlung verwendet werden.

4.5.1.3 Ergänzungsaufgaben

Bei der Auswertung wird für jede Lücke das eingegebene Lösungswort mit dem richtigen bzw. den richtigen Lösungsworten verglichen.

Falls aufgrund der Eingaben bei der Testerstellung Tipp- und Rechtschreibfehler berücksichtigt werden sollen, prüft ggf. ein Algorithmus, ob das Wort mit einem richtigen Lösungswort ausreichend übereinstimmt. Die Punkte der Aufgabe werden im Verhältnis der richtig gelösten Lücken zur Gesamtzahl der Lücken vergeben:

$$\text{Punktvergabe} = \text{erreichbare Punkte} * \frac{\text{richtig gelöste Lücken}}{\text{Anzahl Lücken}} \quad (4.2)$$

4.5.2 Testauswertung

Die Testauswertung gliedert sich in die Ermittlung der Rohwerte aller Teilnehmenden sowie in eine Bewertung der Rohwerte.

4.5.2.1 Ermittlung der Rohwerte

Die Ermittlung des Rohwertes einer Teilnehmerin erfolgt nach der Auswertung aller Aufgaben dieser Teilnehmerin. Die Punkte der Aufgaben werden zusammengezählt. Hier werden ggf. auch negative Punkte von Multiple-Choice-Aufgaben mit Mehrfachnennung berücksichtigt. Wurde eine Korrektur des Rohwerts von Multiple-Choice-Aufgaben ohne Mehrfachnennung bei der Testeingabe gefordert, muß diese nach dem gewählten Verfahren erfolgen. Dazu wird die Anzahl der unbearbeiteten und der falsch gelösten Multiple-Choice-Aufgaben ohne Mehrfachnennung ermittelt. Anschließend wird der ermittelte Rohwert gespeichert.

4.5.2.2 Bewertung der Rohwerte

Es kann entweder eine kriteriumsorientierte oder eine normorientierte Bewertung der Rohwerte erfolgen.

Kriteriumsorientierte Bewertung Für eine kriteriumsorientierte Bewertung muß die Notenskala und der entsprechende Notenschlüssel eingegeben werden. Daraufhin kann eine Liste mit den daraus resultierenden Noten für die einzelnen Teilnehmenden sowie der Durchschnitt erstellt werden.

Normorientierte Bewertung Für die normorientierte Bewertung muß es möglich sein, die zugrundeliegende Norm auszuwählen. Diese kann entweder aus allen Teilnehmenden, die den Test jemals bearbeitet haben, oder allen Teilnehmenden eines Durchgangs bestehen.

Es muß möglich sein, eine Liste mit der Rangordnung, der Prozentrangplatzierung oder dem Standardwert für alle Teilnehmenden zu erstellen.

Für eine Benotung muß die Notenskala sowie die gewünschte Verteilung angegeben werden. Sobald eine Note bekanntgemacht wird, darf sie nicht mehr verändert werden.

Für alle Bewertungsarten gilt, daß es möglich sein muß, die Bewertungskriterien zu verändern oder bekanntzugeben. Wird der Notenschlüssel bekanntgegeben, werden die Noten für die Teilnehmenden gespeichert und müssen von ihnen eingesehen werden können.

4.5.3 Aufgabenanalyse

Bei der Aufgabenanalyse wird der Schwierigkeitsindex und die Trennschärfe einzelner Aufgaben berechnet. Für Multiple-Choice-Aufgaben sollte zusätzlich eine Distraktorenanalyse erfolgen können. Bei mehreren Testdurchläufen muß es möglich sein, die bei der Analyse zu berücksichtigenden Durchläufe auszuwählen.

4.6 Anforderungen zur Sicherheit

Es darf nicht jeder Person möglich sein, Testergebnisse beliebiger anderer Personen in Erfahrung zu bringen. Die Manipulation eines Tests darf nur durch die Person erfolgen, die den Test gestellt hat oder die von dieser Person die Erlaubnis erhält.

Aus diesem Grund dürfen alle Werkzeuge erst nach einer Identifizierung und Authentifizierung durch ein Passwort benutzt werden. Abhängig von der Authentifizierung der Person darf anschließend auf bestimmte Daten zugegriffen werden. Die Teilnehmenden eines Tests dürfen durch das Werkzeug zur Testauswertung nur ihre eigenen Ergebnisse einsehen, wohingegen die Person, die den Test erstellte, alle Rohwerte einsehen darf.

Kapitel 5

Der Datenbankentwurf

In diesem Kapitel wird das Datenbankschema entwickelt und vorgestellt. Zuerst wird ein Entity-Relationship-Modell erstellt und anschließend, ausgehend von diesem Modell, die Relationen der Datenbank abgeleitet. Zum Schluß wird kurz auf die Techniken zur Konsistenzhaltung innerhalb der Datenbank eingegangen.

5.1 Das Entity-Relationship-Modell

Das *Entity-Relationship-Modell* (ER-Modell) der Datenbank entwickelt sich aus den ER-Modellen für die Aufgaben, die Aufgabengebiete, die Tests und die Testinstanzen sowie einem ER-Modell, in dem die Verwendung gleicher Testaufgaben für alle Teilnehmenden eines Testdurchlaufes berücksichtigt wird. Diese Ausschnitte des ER-Modells werden zuerst einzeln beschrieben und anschließend wird das gesamte Datenbankschema vorgestellt. Die Darstellung der ER-Modelle orientiert sich an [Reu93].

5.1.1 Aufgabe

Das ER-Modell des Ausschnitts Aufgabe ist in Abbildung 5.1 dargestellt. Die Entity-Menge *Aufgabe* ergibt sich aus einer Generalisierung der Entity-Mengen *Multiple-Choice-Aufgabe* und *Ergänzungsaufgabe*. Die Generalisierung wird für den weiteren Aufbau des ER-Modells benötigt, indem Aufgaben unabhängig von ihrem Typ betrachtet werden. Multiple-Choice-Aufgaben werden nicht weiter in Aufgaben mit und ohne Mehrfachnennung unterschieden, da sich die Attribute dieser Ausprägungen nicht unterscheiden. Eine Multiple-Choice-Aufgabe enthält mehrere *Items*. Jedes Element der Entity-Menge Item ist genau einer Multiple-Choice-Aufgabe zugeordnet. Jeder Ergänzungsaufgabe können mehrere *Lösungswörter* zugewiesen sein, zum einen durch mehrere Lücken und zum anderen durch mehrere richtige Lösungsmöglichkeiten für eine Lücke. Jedes Lösungswort ist genau einer Ergänzungsaufgabe zugeordnet. Jede Aufgabe besitzt eine Schwierigkeitsstufe und jede Schwierigkeitsstufe kann für mehrere Aufgaben vergeben werden. Die Attribute der Entity-Mengen werden in Tabelle 5.1 beschrieben. Aufgaben werden eindeutig durch die

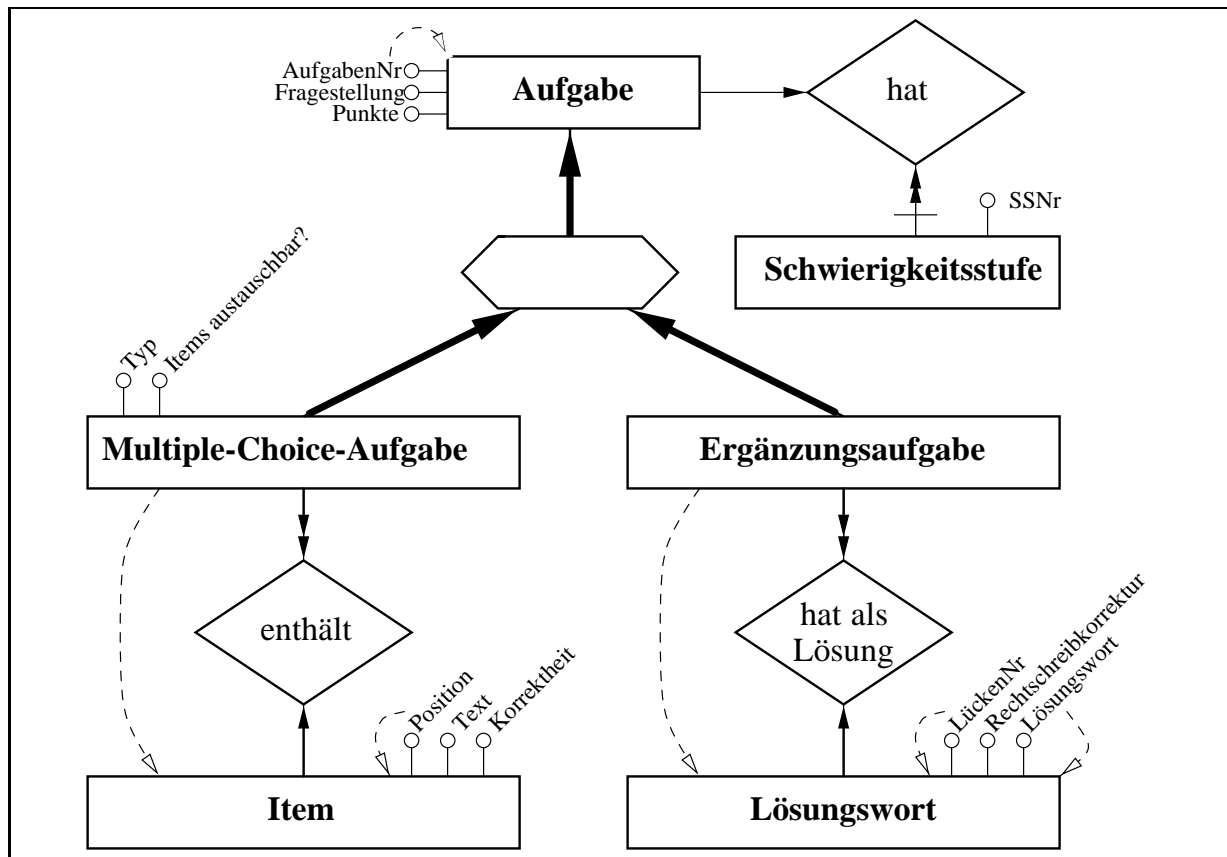


Abbildung 5.1: Das ER-Modell der Aufgabe

Entity-Menge	Attribut	Beschreibung
Aufgabe	AufgabenNr	Dient der eindeutigen Identifizierung
	Fragestellung	Die Fragestellung der Aufgabe
	Punkte	Die in der Aufgabe vergebenen Punkte ^a
Schwierigkeitsstufe	SSNr	Die Schwierigkeitsstufen-Nummer identifiziert die Schwierigkeitsstufe und legt eine Reihenfolge der Stufen fest.
Multiple-Choice-Aufgabe	Typ	Kann die Typen mit Mehrfachnennung und ohne Mehrfachnennung enthalten
	Items austauschbar?	Gibt an, ob die Reihenfolge der Items bei der Aufgabenstellung vertauscht werden darf
Ergänzungsaufgabe	keine eigenen Attribute	
Item	Position	Gibt die Position des Items in der Aufgabe an, falls keine Vertauschung vorgenommen werden darf.
	Text	Der Text des Items
	Korrektheit	Gibt an, ob Item eine richtige oder falsche Lösung ist ^b
Lösungswort	LückenNr	Gibt an, für welche Lücke innerhalb der Aufgabe das Lösungswort gilt
	Lösungswort	Der Text des Lösungsworts
	Rechtschreibkorrektur	Gibt an, ob eine Korrektur der Rechtschreibung bei der Auswertung vorgenommen werden soll

^aIn der Regel ein Punkt, bei Ergänzungsaufgaben für jede Lücke ein Punkt

^bJeder Multiple-Choice-Aufgabe ohne Mehrfachnennung darf nur ein richtiges Item zugewiesen werden

Tabelle 5.1: Attribute der Entity-Mengen des ER-Modells Aufgabe

Aufgaben-Nummer bestimmt, Items durch eine Aufgaben-Nummer und die Position innerhalb der Aufgabe. Lösungsworte werden durch die Aufgaben-Nummer, die Lücke innerhalb der Aufgabe sowie den Wortlaut eindeutig bestimmt. Die SSNr (Schwierigkeitsstufen-Nummer) bestimmt die Schwierigkeitsstufe.

5.1.2 Aufgabengebiet

Das ER-Modell des Ausschnitts Aufgabengebiet ist in Abbildung 5.2 dargestellt. Eine Aufgabe gehört zu einem Aufgabengebiet. Jede Aufgabe kann mehreren Aufgabengebieten zugeordnet werden und jedes Aufgabengebiet kann mehrere Aufgaben enthalten. In jedem Aufgabengebiet können weitere Aufgabengebiete enthalten sein und jedem Aufgabengebiet können andere Aufgabengebiete übergeordnet werden.

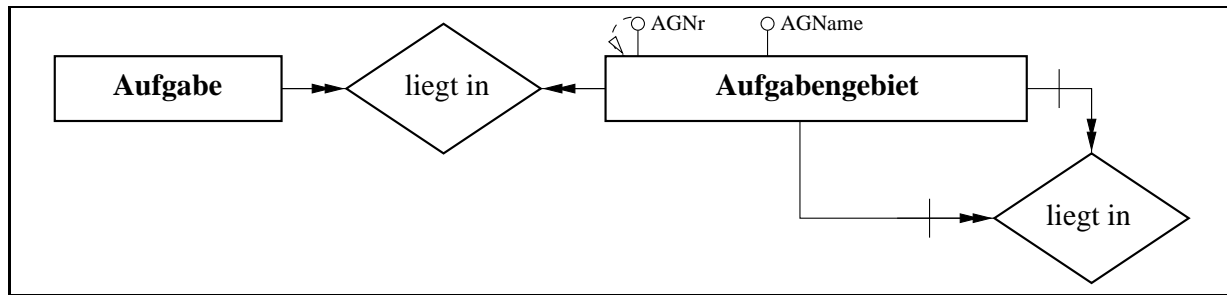


Abbildung 5.2: Das ER-Modell des Aufgabengebiets

Entity-Menge	Attribut	Beschreibung
Aufgabe		bereits in Tabelle 5.1 beschrieben
Aufgabengebiet	AGNummer	Dient der eindeutigen Identifizierung
	AGName	Name des Aufgabengebiets

Tabelle 5.2: Attribute der Entity-Mengen des ER-Modells Aufgabengebiet

In Tabelle 5.2 sind die Attribute der Entity-Mengen beschrieben. Ein Aufgabengebiet wird durch die Aufgabengebiet-Nummer (AGNr) eindeutig beschrieben. Dadurch ist es möglich, daß mehrere Aufgabengebiete mit gleichem Namen existieren. Dies ist beispielsweise sinnvoll, falls verschiedene Personen das Aufgabengebiet mit gleichem Namen erstellen wollen, ohne die Aufgaben der anderen Person zu verwenden.

5.1.3 Test

Abbildung 5.3 zeigt die Darstellung des ER-Modells Test. Die Entity-Menge *Test* ist eine Generalisierung der Entity-Mengen *Test mit Aufgabengenerierung* und *CAT-Test*. Ein Test mit Aufgabengenerierung enthält mehrere Aufgabengebiete (mindestens eins). Jeder Test hat genau eine Testanleitung die kann für mehrere Tests verwendet werden kann. Für jeden Test mit Aufgabengenerierung und jedes in diesem Test enthaltene Aufgabengebiet muß für jede verwendete Schwierigkeitsstufe angegeben werden, wieviel solcher Aufgaben im Test verwendet werden sollen. Die Angabe, wieviel Aufgaben verwendet werden, erfolgt nicht absolut, sondern über die vergebene Punktzahl. Ausgehend von dieser Punktzahl werden Aufgaben gewählt, die insgesamt diese Punktzahl ergeben.

Die Schwierigkeitsstufen der Aufgaben werden als eigene Entity-Menge *Schwierigkeitsstufe* eingeführt, um die Wertemenge explizit zu modellieren.

Ein CAT-Test enthält mehrere Aufgabengebiete (mindestens eins). Für jeden CAT-Test und jedes darin enthaltenen Aufgabengebiet muß die Anzahl der Aufgaben (indirekt über die Punktzahl), die Schrittweite bis zur Überprüfung der Schwierigkeitsstufe (ebenfalls indirekt über Punktzahl), sowie die Kriterien zur Änderung der Schwierigkeitsstufe nach oben (Attribut: ÄK no) und nach unten (Attribut: ÄK nu) angegeben werden. Für jeden CAT-Test und jedes darin enthaltenen Aufgabengebiet muß zusätzlich angegeben werden,

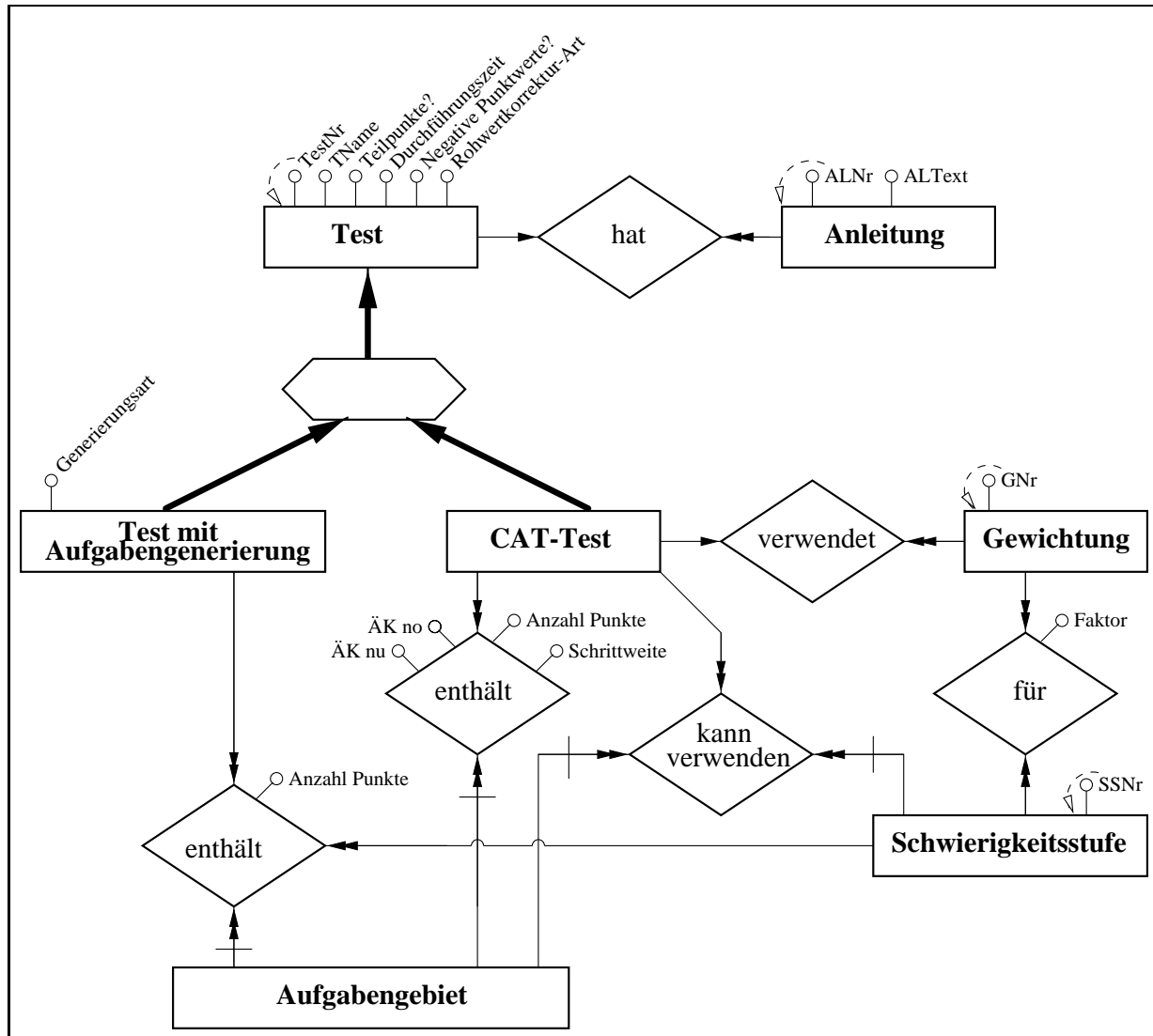


Abbildung 5.3: Das ER-Modell des Tests

Entity-Menge	Attribut	Beschreibung
Test	TestNr	Identifiziert den Test eindeutig
	TName	Name des Tests
	Durchführungszeit	Die Zeit, die ein Kandidat zur Bearbeitung des Tests hat
	Negative Punktwerte?	Gibt an, ob bei Multiple-Choice-Aufgaben mit Mehrfachnennung negative Punktwerte über die Aufgabengrenze vergeben werden
	Rohwertkorrektur-Art	Gibt die Art der Rohwertkorrektur für Multiple-Choice-Aufgaben ohne Mehrfachnennung an
	Teilpunkte?	Bestimmt, ob für teilweise richtige Lösungen Teilpunkte vergeben werden
Test mit Aufgabengenerierung	Generierungsart	Gibt an, ob die generierten Aufgaben für alle Teilnehmenden gleich oder verschieden sind
CAT-Test		keine eigenen Attribute
Gewichtung	GNr	Eindeutige Identifizierung
Schwierigkeitsstufe	SSNr	Eindeutige Identifizierung
Aufgabengebiet		bereits in Tabelle 5.2 beschrieben
Anleitung	ALNr	Dient der Identifizierung
	ALText	Der Text der Testanleitung, der allen Teilnehmenden vor Testbeginn angezeigt wird

Tabelle 5.3: Attribute der Entity-Mengen des ER-Modells Test

welche Schwierigkeitsstufen verwendet werden können. Der CAT-Test startet bei jedem Aufgabengebiet immer in der mittleren der gewählten Schwierigkeitsstufen. Falls eine gerade Anzahl an Schwierigkeitsstufen gewählt wurde, wird die niedrigere der beiden mittleren Schwierigkeitsstufen gewählt. Zur Ermittlung des Rohwerts verwendet ein CAT-Test für jede Aufgabe, abhängig von der Schwierigkeitsstufe, einen Faktor, mit dem der Punktwert der Aufgabe multipliziert wird. Für die mittlere Schwierigkeitsstufe wird in der Regel der Faktor eins gewählt. Für höhere Schwierigkeitsstufen wird ein Faktor größer eins, für niedrigere Schwierigkeitsstufen ein Faktor kleiner eins verwendet¹. Diese Faktoren sind in der Entity-Menge *Gewichtung* gespeichert. Jeder CAT-Test verwendet eine Gewichtung, in der für jede Schwierigkeitsstufe ein Faktor festgelegt ist. Eine Gewichtung kann von mehreren CAT-Tests verwendet werden. Die Beschreibung der einzelnen Attribute der Entity-Mengen erfolgt in Tabelle 5.3.

¹Damit erhalten die Kandidaten, die schwierigere Aufgaben bearbeiten, mehr Punkte

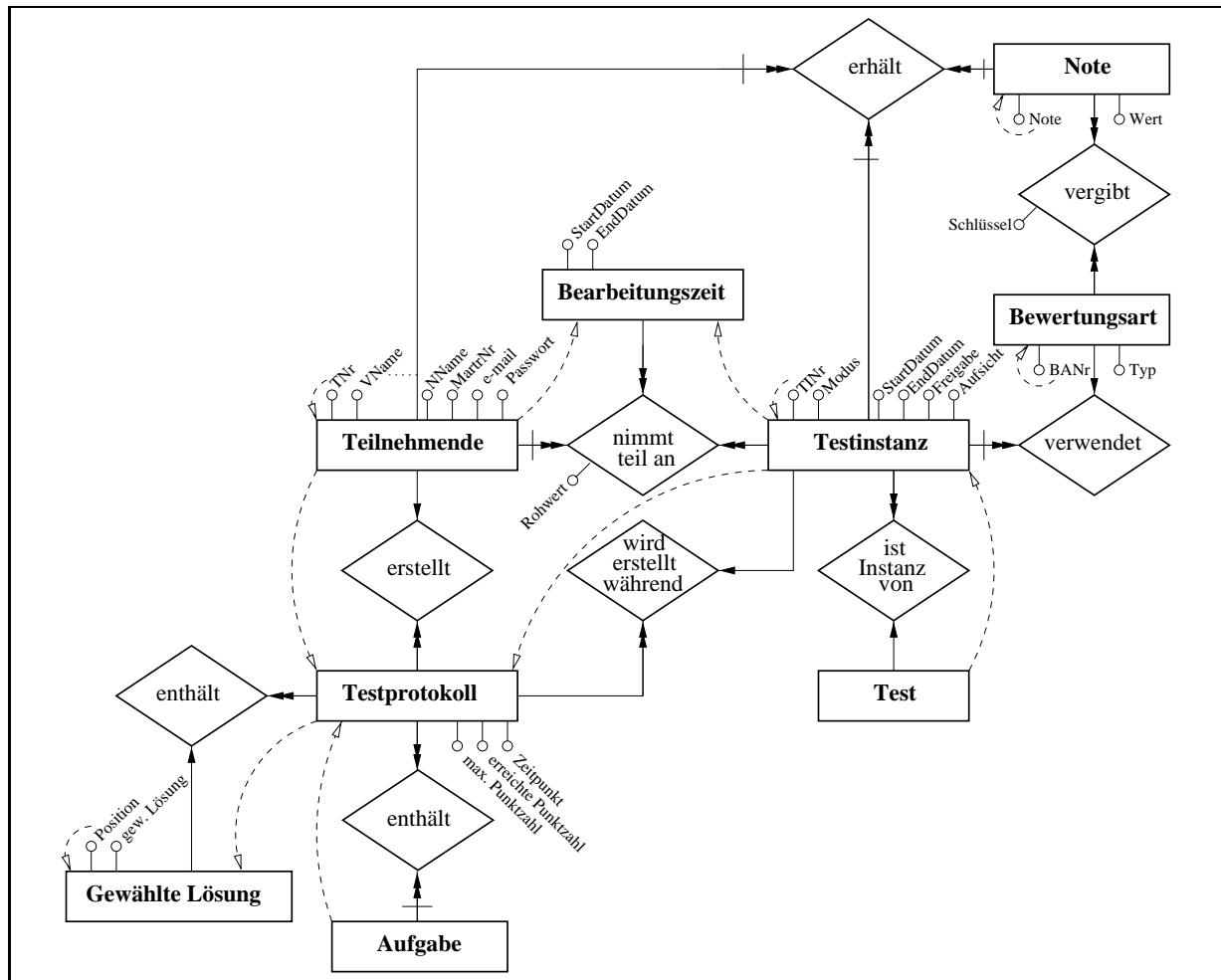


Abbildung 5.4: Das ER-Modell der Testinstanz

5.1.4 Testinstanz

In Abbildung 5.4 ist das ER-Modell des Ausschnitts Testinstanz zu sehen. In der Entity-Menge *Testinstanz* sind die einzelnen Durchführungen der Tests gespeichert. Jeder Test kann mehrere Testinstanzen besitzen. Jede Testinstanz ist genau einem Test zugeordnet. Jeder Testinstanz sind *Teilnehmende* zugeordnet. Jede Teilnehmende kann an mehreren Testinstanzen teilnehmen. Jeder Teilnehmende erhält innerhalb jeder Testinstanz, an der er teilnimmt, einen Rohwert. Die Bearbeitungszeit eines Teilnehmenden in einer Testinstanz wird in der Entity-Menge *Bearbeitungszeit* gespeichert. Die Bearbeitungszeit wird in einer eigenen Menge gespeichert, da ein Ausfall des Testprogramms dazu führen kann, dass eine Teilnehmende für eine Testinstanz mehrere Bearbeitungszeiten haben kann. Für jede Aufgabe jedes Teilnehmenden jeder Testinstanz wird ein *Testprotokoll* erstellt. In ihm werden die erreichten Punkte sowie die maximal erreichbaren Punkte gespeichert. Zusätzlich wird die vom Teilnehmenden gewählte Lösung in der Entity-Menge *Gewählte Lösung*

Entity-Menge	Attribut	Beschreibung
Testinstanz	TINr	Gibt an, die wievielte Ausführung eines Tests die Testinstanz darstellt
	StartDatum	Ab wann darf Testinstanz bearbeitet werden
	EndDatum	Bis wann muß Testinstanz bearbeitet werden
	Modus	Gibt an, ob Aufgaben aufgrund vorheriger Ergebnisse generiert werden ^a
	Freigabe	Bestimmt, ob und wann die richtigen Lösungen der Aufgaben den Teilnehmenden bekanntgegeben werden
	Aufsicht	Gibt an, ob eine Aufsicht erforderlich ist ^b
Test	bereits in Tabelle 5.3 beschrieben	
Aufgabe	bereits in Tabelle 5.1 beschrieben	
Bearbeitungszeit	StartDatum	Gibt an, wann Bearbeitung begonnen wurde
	EndDatum	Gibt an, wann Bearbeitung beendet wurde
Teilnehmende	TNr	Identifiziert die Teilnehmenden eindeutig
	VName	Vorname
	NName	Nachname
	Passwort	Passwort zur Identifizierung
	MatrNr	Matrikelnummer für Studierende
	e-mail	E-Mail-Adresse
	weitere Attribute möglich	
Testprotokoll	erreichte Punktzahl	Die vom Teilnehmenden erreichte Punktzahl
	max. Punktzahl	Maximal vergebene Punktzahl ^c
	Zeitpunkt	Zeitpunkt der Beantwortung
Gewählte Lösung	Position	Die Position der Lösung innerhalb der Aufgabe
	gew. Lösung	Von der Teilnehmenden gewählte Lösung ^d
Note	Note	Der Name der Note (kann Zahl oder Text sein)
	Wert	Legt die Reihenfolge der Noten fest
Bewertungsart	BANr	Identifiziert die Bewertungsart
	Typ	Gibt an, ob eine norm- oder kriteriumsorientierte Bewertung vorgenommen wird

^aIm Übungsmodus sollten vom Teilnehmenden nicht richtig beantwortete Aufgaben öfter gestellt werden, oft richtig beantwortete Aufgaben sollten nur selten gestellt werden. Im Testmodus dürfen vorherige Ergebnisse keine Rolle spielen und, falls möglich, nur vom Teilnehmenden noch nicht bearbeitete Aufgaben gestellt werden.

^bFalls eine Aufsicht erforderlich ist, wird das entsprechende Passwort gespeichert, ansonsten wird dieses Attribut nicht verwendet

^cbesonders wichtig, falls bei CAT-Tests aufgrund der Gewichtung der Schwierigkeitsstufen die Punktzahl verändert wird

^dbei Ergänzungsaufgaben das an dieser Stelle eingegebene Wort, bei Multiple-Choice-Aufgaben die Item-Nummer

Tabelle 5.4: Attribute der Entity-Mengen des ER-Modells Test

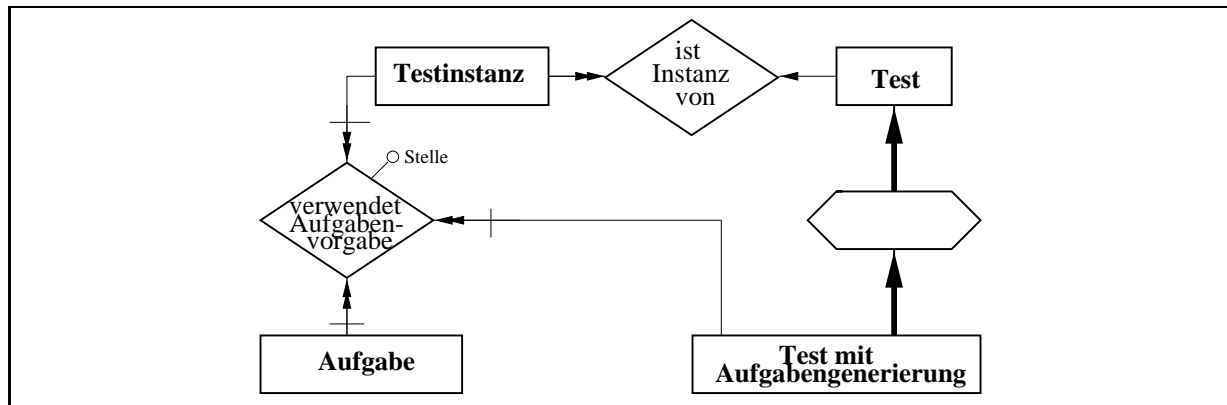


Abbildung 5.5: Das ER-Modell, mit dem alle Teilnehmenden eines Testdurchlaufs die gleichen Aufgaben erhalten

gespeichert (auch mehrere Antworten innerhalb einer Aufgabe). Die Bewertungskriterien einer Testinstanz werden in der Entity-Menge *Bewertungsart* gespeichert. Jede Testinstanz besitzt höchstens eine Bewertungsart und jede Bewertungsart kann von mehreren Testinstanzen verwendet werden. Die Bewertungsart bestimmt über einen Notenschlüssel, welche *Note* bei welchem Rohwert vergeben wird. Jeder Teilnehmende kann für jede Testinstanz eine Note erhalten. In Tabelle 5.4 sind die Attribute der Entity-Mengen beschrieben.

5.1.5 Gleiche Aufgaben für alle Teilnehmenden eines Test-Durchlaufs

Damit alle Teilnehmenden einer Testinstanz von einem Test mit Aufgabengenerierung die gleichen Aufgaben erhalten, wird für jede Testinstanz des Tests für alle verwendeten Aufgaben festgelegt, in welcher Reihenfolge sie gestellt werden. Abbildung 5.5 zeigt das entsprechende ER-Modell. Die Attribute der Entity-Mengen wurden bereits in den vorherigen Abschnitten beschrieben.

5.1.6 Das Datenbankschema

Aus den beschriebenen ER-Modellen ergibt sich für die Datenbank ein ER-Modell. Dieses ist in Abbildung 5.6 dargestellt. Im folgenden werden Abhängigkeiten beschrieben, die in den bisher beschriebenen ER-Modellen noch nicht erwähnt wurden. CAT-Tests dürfen lediglich dann abgespeichert werden, wenn für jedes Aufgabengebiet und jede in diesem Aufgabengebiet verwendete Schwierigkeitsstufe die geforderte Punktzahl mit den vorhandenen Aufgaben erreicht werden kann. Tests mit Aufgabengenerierung können dann abgespeichert werden, wenn die angegebene Punktzahl für die gewählten Schwierigkeitsstufen und Aufgabengebiete mit den vorhandenen Aufgaben erreicht werden kann. Das Verändern der Schwierigkeitsstufe, des Aufgabengebiets oder der Punktzahl sowie das Löschen einer Aufgabe darf nur erfolgen, wenn für keinen Test die obigen Bedingungen verletzt werden.

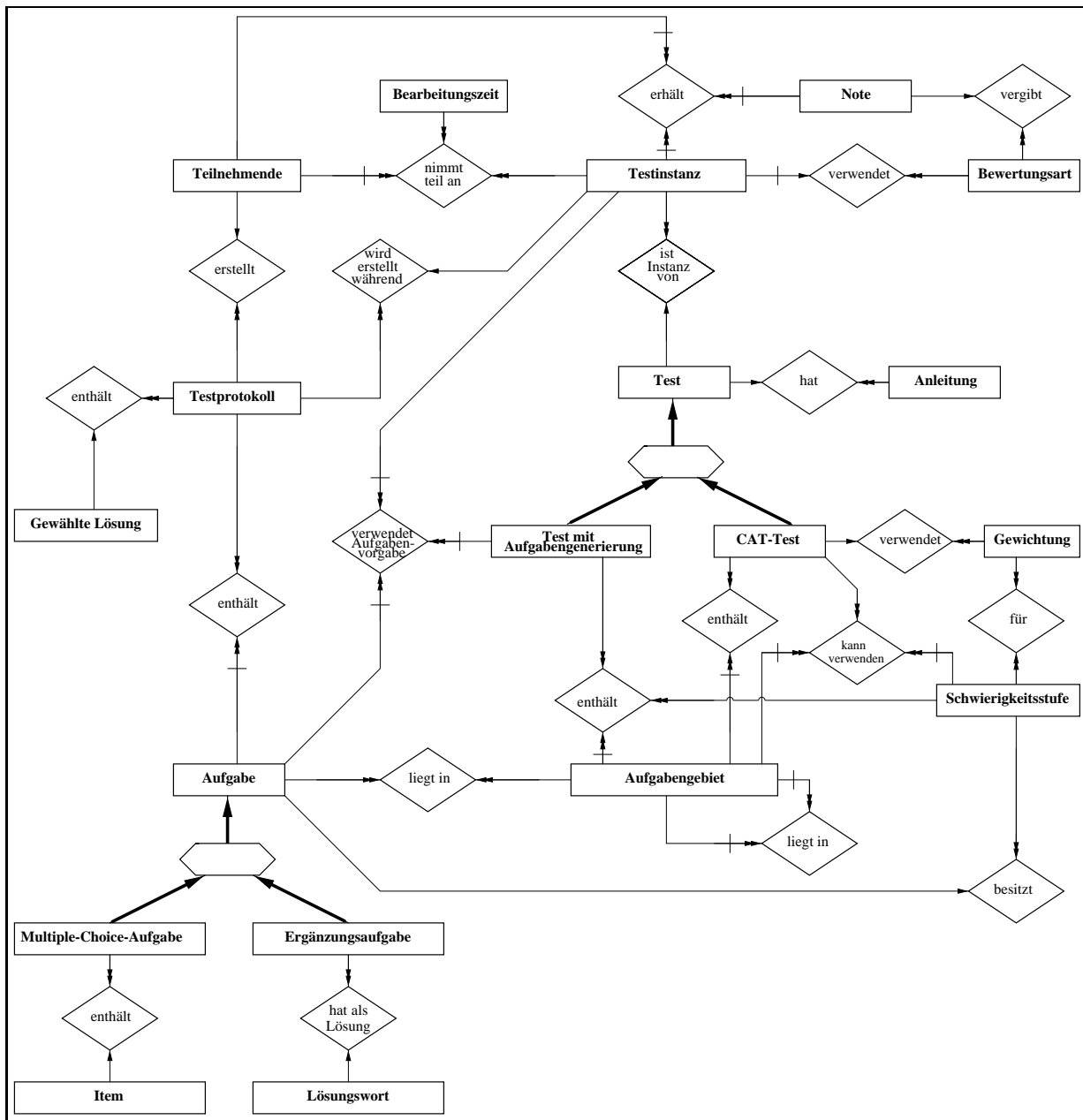


Abbildung 5.6: Das ER-Modell der Datenbank

5.2 Die Relationen

Im folgenden werden die Relationen beschrieben, die sich aus dem ER-Modell ergeben. Dazu werden der Primärschlüssel sowie die Fremdschlüssel und für jedes Attribut einer Relation der Wertebereich festgelegt.

5.2.1 WWWT_Aufgabe

WWWT_Aufgabe²(AufgabenNr, Fragestellung, Punkte, Typ, Items_austauschbar ³, SSNr)

Die Relation **WWWT_Aufgabe** ergibt sich aus den Entity-Mengen Multiple-Choice-Aufgabe und Ergänzungsaufgabe und deren Generalisierung, der Aufgabe. Die Schwierigkeitsstufe einer Aufgabe wird ebenfalls in dieser Relation gespeichert.

Jede Aufgabe vom Typ Multiple-Choice muß mindestens ein (mit Mehrfachnennung) bzw. zwei (ohne Mehrfachnennung) Items enthalten, jede Aufgabe vom Typ Ergänzungsaufgabe für jede Lücke mindestens ein Lösungswort.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AufgabenNr	INTEGER NOT NULL	Ja	
Fragestellung	VARCHAR(2000) NOT NULL ^a		
Punkte	INTEGER NOT NULL ^b		
Typ	VARCHAR(5) NOT NULL ^c		
Items_austauschbar	CHAR(1) ^d		
SSNr	INTEGER NOT NULL		WWWT_Schw_Stufe

^aFür Ergänzungsaufgaben muß die Fragestellung Platzhalter für die Lücken enthalten

^bBedingung: Wert > 0

^cKann die Werte 'MCMMN' (Multiple-Choice-Aufgabe ohne Mehrfachnennung), 'MCOMN' (Multiple-Choice-Aufgabe mit Mehrfachnennung) und 'EGA' (Ergänzungsaufgabe) annehmen

^dWird nur bei Multiple-Choice-Aufgaben verwendet. Kann die Werte 'T' und 'F' annehmen.

5.2.2 WWWT_Schw_Stufe

WWWT_Schw_Stufe(SSNr)

Die Entity-Menge Schwierigkeitsstufe wird auf die Relation **WWWT_Schw_Stufe** abgebildet.

²Die Relationen werden mit dem Namen beschrieben, die bei der Realisierung verwendet werden. Der Präfix 'WWWT_' steht für World Wide Web basiertes Testen

³Das verwendete Datenbanksystem (DB2 V5.0) unterstützt Relationen- und Attributnamen von max. 18 Zeichen Länge. Deshalb werden die Namen abweichend vom ER-Modell gekürzt, falls der Name (bei den Relationen inklusive Präfix) länger als 18 Zeichen ist.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
SSNr	INTEGER NOT NULL ^a	Ja	

^aJe größer der Wert, um so schwieriger ist die Aufgabe

5.2.3 WWWT_Item

WWWT_Item(AufgabenNr, Position, Text, Korrektheit)

Die Relation **WWWT_Item** ergibt sich aus der Entity-Menge **Item** und enthält zur eindeutigen Identifizierung den Primärschlüssel der Relation **WWWT_Aufgabe**. Es dürfen nur **AufgabenNr** von Aufgaben enthalten sein, die vom Typ **MCOMN** oder **MCMMN** sind.

Für jede Aufgabe vom Typ **MCOMN** darf nur von einem **Item** das Attribut **Korrektheit** den Wert **True (T)** besitzen.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
Position	INTEGER NOT NULL	Ja	
Text	VARCHAR(254) NOT NULL		
Korrektheit	CHAR(1) NOT NULL ^a		

^aKann die Werte 'T' und 'F' annehmen

5.2.4 WWWT_Loesungswort

WWWT_Loesungswort(AufgabenNr, LueckenNr, Loesungswort, Rechtschreibkorr)

Die Entity-Menge **Lösungswort** wird auf die Relation **WWWT_Lösungswort** abgebildet, wobei der Primärschlüssel der Relation **WWWT_Aufgabe** übernommen wird. Es dürfen nur **AufgabenNr** von Aufgaben mit Typ **EGA** enthalten sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
LueckenNr	INTEGER NOT NULL ^a	Ja	
Loesungswort	VARCHAR(50) NOT NULL	Ja	
Rechtschreibkorr	CHAR(1) NOT NULL ^b		

^aBestimmt die Lücke innerhalb der Fragestellung. Die Werte müssen zwischen eins und der Anzahl der Lücken liegen.
^bkann die Werte 'T' oder 'F' annehmen

5.2.5 WWWT_AG

WWWT_AG(AGNr, AGName)

Die Relation entsteht aus der Entity-Menge Aufgabengebiet.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AGNr	INTEGER NOT NULL	Ja	
AGName	VARCHAR(50) NOT NULL		

5.2.6 WWWT_Au_liegtin_AG

WWWT_Au_liegtin_AG(AufgabenNr, AGNr)

Diese Relation entsteht aus der Relationship “liegt in” zwischen Aufgabe und Aufgabengebiet. Sie enthält außer den Primärschlüsseln der beiden in Beziehung stehenden Entity-Mengen keine Attribute.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
AGNr	INTEGER NOT NULL	Ja	WWWT_AG

5.2.7 WWWT_AG_liegtin_AG

WWWT_AG_liegtin_AG(AGNr_liegt_in, AGNr_enthaelt)

Aus der Relationship “liegt in” zwischen Aufgabengebiet und Aufgabengebiet ergibt sich diese Relation. Das erste Attribut bestimmt das untergeordnete Aufgabengebiet, das zweite Attribut das übergeordnete Aufgabengebiet. Es wird nicht verhindert, daß sich zwei Aufgabengebiete enthalten. In diesem Fall ist die Menge der Aufgaben beider Aufgabengebiete gleich.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
AGNr_liegt_in	INTEGER NOT NULL	Ja	WWWT_AG
AGNr_enthaelt	INTEGER NOT NULL	Ja	WWWT_AG

5.2.8 WWWT_Test

WWWT_Test(TestNr, TName, Teilpunkte_erlaubt, Typ, Durchfuehrungszeit, Rohwertkorrektur, Neg_Punktwerte, Generierungsart, ALNr, GNr)

Aus den Entity-Mengen Test mit Aufgabengenerierung und CAT-Test sowie deren Generalisierung Test entsteht die Relation **WWWT_Test**. Der Anleitungstext wird in einer eigenen Relation gespeichert und ist über den zu diesem Zweck eingeführten Schlüssel **ALNr** zu identifizieren. Damit wird erreicht, daß der Text der Anleitung nicht in der Relation **WWWT_Test** gespeichert wird. Da die Anleitungen für mehrere Tests verwendet werden können, würde der gleiche Text sonst mehrfach abgespeichert werden.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TestNr	INTEGER NOT NULL	Ja	
TName	VARCHAR(50) NOT NULL		
Teilpunkte_erlaubt	CHAR(1) NOT NULL ^a		
Typ	VARCHAR(3) NOT NULL ^b		
Durchfuehrungszeit	TIME NOT NULL		
Rohwertkorrektur	VARCHAR(3) ^c		
Neg_Punktwerte	CHAR(1) NOT NULL ^a		
Generierungsart	VARCHAR(2) ^d		
ALNr	INTEGER NOT NULL		WWWT_Anleitung
GNr	INTEGER ^e		WWWT_Gewichtung

^akann die Werte 'T' und 'F' annehmen
^bKann die Werte 'CAT' und 'GEN' annehmen
^ckann die Werte 'AP' (Abzug der Punkte), 'ATP' (Abzug Teilpunkte) oder 'ZTP' (Zuzählen von Teilpunkten) annehmen
^dWird nur bei Tests des Typs GEN verwendet. Kann die Werte 'FA' (für alle eine Generierung) oder 'FJ' (für jeden Teilnehmenden eine Generierung) annehmen.
^eWird nur bei Tests des Typs CAT verwendet

5.2.9 WWWT_Anleitung

WWWT_Anleitung(ALNr, ALText)

Diese Relation speichert die Anleitung eines Tests. Da die Testanleitung viele Zeichen enthalten kann, wird sie nicht bei der Relation Test gespeichert. Eine Testanleitung kann für mehrere Tests verwendet werden und würde sonst mehrfach abgespeichert werden.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
ALNr	INTEGER NOT NULL	Ja	
ALText	VARCHAR(3997) ^a		

^aDies ist die maximale Größe, die in dem verwendeten Datenbanksystem (DB/2 V5.0) erlaubt ist (Pro Zeile max. 4005 Zeichen, ALNr nimmt 8 Zeichen ein). Sollten die Testanleitungen länger sein, müßte als Datentyp ein CLOB (Character Large Object) verwendet werden (siehe [Cha96]). Meiner Meinung nach reichen knapp 4000 Zeichen für eine kurze und prägnante Testanleitung aus, weshalb auf die Verwendung von CLOBs verzichtet wird.

5.2.10 WWWT_GenT_enth_AG

WWWT_GenT_enth_AG(TestNr, AGNr, SSNr, Anzahl_Punkte)

Die Relation WWWT_GenT_enth_AG entsteht aus der Relationship-Menge “enthält” zwischen Test mit Aufgabengenerierung und Aufgabengebiet. Die Primärschlüssel dieser beiden Entity-Mengen werden übernommen. Es dürfen nur Tests vom Typ GEN enthalten sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
AGNr	INTEGER NOT NULL	Ja	WWWT_AG
SSNr	INTEGER NOT NULL	Ja	WWWT_Schw_Stufe
AnzahlPunkte	DECIMAL(10,2) NOT NULL		

5.2.11 WWWT_CAT_enth_AG

WWWT_CAT_entha_AG(TestNr, AGNr, AEK_nu, AEK_no, Anzahl_Punkte, Schrittweite)

Die Relationship-Menge “enthält” zwischen den Entity-Mengen CAT-Test und Aufgabengebiet ergibt die Relation WWWT_CAT_enth_AG, die die Primärschlüssel der beiden Entity-Mengen übernimmt. Es dürfen nur Tests vom Typ CAT enthalten sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
AGNr	INTEGER NOT NULL	Ja	WWWT_AG
AEK_nu	DECIMAL(10,2) NOT NULL		
AEK_no	DECIMAL(10,2) NOT NULL		
AnzahlPunkte	DECIMAL(10,2) NOT NULL		
Schrittweite	DECIMAL(10,2) NOT NULL		

5.2.12 WWWT_CAT_kann_verw

WWWT_CAT_kann_verw(TestNr, AGNr, SSNr)

Diese Relation ergibt sich aus der Relationship “kann verwenden” zwischen den Entity-Mengen CAT-Test, Schwierigkeitsstufe und Aufgabengebiet. Die Primärschlüssel der drei Entity-Mengen werden übernommen. Wie bei der Relation WWWT_CAT_enth_AG dürfen nur Tests vom Typ CAT enthalten sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
AGNr	INTEGER NOT NULL	Ja	WWWT_AG
SSNr	INTEGER NOT NULL	Ja	WWWT_Schw_Stufe

5.2.13 WWWT_Gewichtung

WWWT_Gewichtung(GNr)

Die Relation WWWT_Gewichtung entsteht aus der Entity-Menge Gewichtung.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
GNr	INTEGER NOT NULL	Ja	

5.2.14 WWWT_Gew_fuer_SS

WWWT_Gew_fuer_SS(GNr, SSNr, Faktor)

Die Relation WWWT_Gew_für_SS ergibt sich aus der Relationship-Menge “für” zwischen Gewichtung und Schwierigkeitsstufe, enthalten sind die Primärschlüssel beider Entity-Mengen.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
GNr	INTEGER NOT NULL	Ja	WWWT_Gewichtung
SSNr	INTEGER NOT NULL	Ja	WWWT_Schw_Stufe
Faktor	DECIMAL(2,1) NOT NULL		

5.2.15 WWWT_Teilnehmende

WWWT_Teilnehmende(TNr, VName, NName, MatrNr, e-mail, Passwort)

Aus der Entity-Menge Teilnehmende wird die Relation WWWT_Teilnehmende. Diese Relation kann durch weitere Attribute ergänzt werden. Insbesondere sollten bei einer Integration des entworfenen Datenbankschemas in eine Lehrumgebung abgespeichert werden, welche Kurse die Teilnehmenden besuchen.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TNr	INTEGER NOT NULL	Ja	
VName	VARCHAR(50)		
NName	VARCHAR(50)		
MatrNr	INTEGER		
e_mail	VARCHAR(50)		
Passwort	VARCHAR(10)		

5.2.16 WWWT_Testinstanz

WWWT_Testinstanz(TINr, TestNr, Start_Datum, Start_Zeit, End_Datum, End_Zeit, Aufsicht, BANr, Freigabe)

Die Relation **WWWT_Testinstanz** entsteht aus der Entity-Menge **Testinstanz**. Die Relation enthält den Primärschlüssel der Relationen **WWWT_Test** und **WWWT_Bewertungsart**.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TINr	INTEGER NOT NULL ^a	Ja	
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
Start_Datum	DATE NOT NULL		
Start_Zeit	TIME NOT NULL		
End_Datum	DATE NOT NULL		
End_Zeit	TIME NOT NULL		
Modus	CHAR(1) NOT NULL ^b		
Aufsicht	VARCHAR(10) ^c		
BANr	INTEGER NOT NULL		WWWT_Bewertungsart
Freigabe	VARCHAR(3) ^d		

^awird für jeden Test um eins inkrementiert

^bKann die Werte 'U' (Übungsmodus) und 'T' (Testmodus) annehmen

^cfalls eine Aufsicht erforderlich ist, wird hier das Passwort der Aufsicht gespeichert, ansonsten NULL

^dKann die Werte 'NIE' (keine Bekanntgabe der Ergebnisse), 'SOF' (sofortige Freigabe der Lösung nach der Aufgabe), 'BTT' (Freigabe nach Beendigung der Testinstanz durch den Teilnehmenden), 'BTA' (Freigabe nach Beendigung der Testinstanz durch alle Teilnehmenden, genauer nach dem Ablauf des Enddatums) annehmen

5.2.17 WWWT_Bewertungsart

WWWT_Bewertungsart(BANr, Typ)

Die Entity-Menge **Bewertungsart** ergibt die Relation **WWWT_Bewertungsart**.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
BANr	INTEGER NOT NULL	Ja	
BATyp	VARCHAR(6) NOT NULL ^a		

^aKann die Werte 'KRIT', 'NORMTI' und 'NORMT' enthalten. KRIT steht für eine kriteriumsorientierte Bewertung, NORMTI für eine normorientierte Bewertung, die die Testinstanz als Norm verwendet. Bei NORMT werden alle bisherigen Testinstanzen des Tests als Norm verwendet.

5.2.18 WWWT_Note

WWWT_Note(Note, Wert)

Die Relation WWWT_Note entsteht aus der Entity-Menge Note.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
Note	VARCHAR(50) NOT NULL ^a	Ja	
Wert	INTERGER NOT NULL ^b		

^aDamit kann die Note sowohl einen Zahlenwert als auch eine Beurteilung (z.B. "nicht bestanden") enthalten

^bWird benötigt, um den Notendurchschnitt berechnen zu können

5.2.19 WWWT_Notenschl

WWWT_Notenschl(BANr, Note, Schluessel)

Die Relation WWWT_Notenschl ergibt sich aus der Relationship "vergift" zwischen Note und Bewertungsart.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
BANr	INTEGER NOT NULL	Ja	WWWT_Bewertungsart
Note	VARCHAR(50) NOT NULL	Ja	WWWT_Note
Schluessel	DECIMAL(5,2) NOT NULL ^a		

^aJe nach Bewertungsart gibt der Schlüssel an, wieviel Prozent der Teilnehmenden diese Note erhalten sollen (normorientiert), oder wieviel Prozent der Punkte für diese Note erforderlich sind (kriteriumsorientiert).

5.2.20 WWWT_Teilnahme

WWWT_Teilnahme(TNr, TINr, TestNr, Rohwert, Note)

WWWT_Teilnahme entsteht aus den Relationships "nimmt teil an" und "erhält" zwischen Teilnehmende und Testinstanz, bzw. zusätzlich Note.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TNr	INTEGER NOT NULL	Ja	WWWT_Teilnehmende
TINr	INTEGER NOT NULL	Ja	WWWT_Testinstanz
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
Rohwert	DECIMAL(10,2)		
Note	VARCHAR(50)		WWWT_Note

5.2.21 WWWT_BearbeitungsZ

WWWT_BearbeitungsZ(TNr, TINr, TestNr, Start_Zeit, End_Zeit)

WWWT_BearbeitungsZ entsteht aus der Entity-Menge Bearbeitungszeit. Es ist möglich, hier mehrere Bearbeitungszeiten für die Teilnehmerin einer Testinstanz zu speichern. Dies ergibt sich aus einem möglichen Ausfall des Testprogramms. Die Information, wie lange die Bearbeitung unterbrochen wurde, könnte bei nur einer Start- und End-Zeit nicht gespeichert werden. Beim Start des Testprogramms wird lediglich die Start_Zeit eingetragen. Sobald eine Aufgabe bearbeitet wurde, wird die aktuelle Zeit in die End_Zeit eingetragen. Die Start_Zeit muß immer kleiner oder gleich der End_Zeit sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TNr	INTEGER NOT NULL	Ja	WWWT_Teilnehmende
TINr	INTEGER NOT NULL	Ja	WWWT_Testinstanz
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
Start_Zeit	TIMESTAMP NOT NULL	Ja	
End_Zeit	TIMESTAMP		

5.2.22 WWWT_Testprotokoll

WWWT_Testprotokoll(TNr, AufgabenNr, TINr, TestNr, Zeitpunkt, max_Punktzahl, erreichte_Punkt_Z)

WWWT_Testprotokoll ergibt sich aus der Entity-Menge Testprotokoll und enthält die Primärschlüssel von WWWT_Teilnehmende, WWWT_Testinstanz und WWWT_Aufgabe.

Attribut	Wertebereich	Primärschl.	Fremdschlüssel
TNr	INTEGER NOT NULL	Ja	WWWT_Teilnehmende
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
TINr	INTEGER NOT NULL	Ja	WWWT_Testinstanz
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
Zeitpunkt	TIMESTAMP		
max_Punktzahl	DECIMAL(10,2) NOT NULL		
erreichte_Punkt_Z	DECIMAL(10,2) ^a		

^aNULL, falls nicht bearbeitet

5.2.23 WWWT_Gew_Loesung

WWWT_Gew_Loesung(TNr, AufgabenNr, TINr, TestNr, Position, gew_Loesung)

Die Entity-Menge Gewählte_Lösung wird zur Relation WWWT_Gew_Loesung und enthält den Primärschlüssel von WWWT_Testprotokoll.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TNr	INTEGER NOT NULL	Ja	WWWT_Teilnehmende
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
TINr	INTEGER NOT NULL	Ja	WWWT_Testinstanz
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
Position	INTEGER NOT NULL	Ja	
gew_Loesung	VARCHAR(50) NOT NULL		

5.2.24 WWWT_Aufg_vorgabe

WWWT_Aufg_vorgabe(TINr, TestNr, AufgabenNr, Stelle)

Die Relation **WWWT_Aufg_vorgabe** entsteht aus der Relationship-Menge “verwendet Aufgabenvorgabe” zwischen Test mit Aufgabengenerierung, Testinstanz und Aufgabe. Es dürfen nur Tests vom Typ GEN enthalten sein.

Attribut	Wertebereich	Primärschlüssel	Fremdschlüssel
TINr	INTEGER NOT NULL	Ja	WWWT_Testinstanz
TestNr	INTEGER NOT NULL	Ja	WWWT_Test
AufgabenNr	INTEGER NOT NULL	Ja	WWWT_Aufgabe
Stelle	INTEGER NOT NULL		

5.3 Durchgeführte Optimierungen

Aus der Notwendigkeit, alle Aufgaben eines bestimmten Aufgabengebietes zu bestimmen, ergibt sich aufgrund der Beziehung “*Aufgabengebiet liegt in Aufgabengebiet*” ein Problem. Vom Datenbank-Entwurf aus betrachtet ist es nicht notwendig, daß für ein Aufgabengebiet AG1 jede enthaltene Aufgabe Ax in der Relation **WWWT_Au_liegtin_AG** mit einem Tupel (Ax, AG1) vermerkt ist. Falls ein anderes Aufgabengebiet AG2 in AG1 enthalten ist, reicht es aus, in **WWWT_Au_liegtin_AG** einen Tupel (Ax, AG2) zu speichern. Zur Bestimmung aller Aufgaben des Aufgabengebiets AG1 ist also zunächst eine Anfrage auf **WWWT_AG_liegtin_AG** erforderlich, die alle Aufgabengebiete liefert, die AG1 enthält. Da diese Aufgabengebiete wiederum andere Aufgabengebiete enthalten können, ist prinzipiell eine rekursive SQL-Anfrage erforderlich. Eine SQL-Anfrage, die alle Aufgaben eines Aufgabengebietes AG1 liefert, könnte folgendermaßen aussehen:

```
WITH zwischenspeicher(AGNr_liegt_in,AGNr_enthaelt,i) AS
  ((SELECT AGNr_liegt_in,AGNr_enthaelt,1 FROM WWWT_AG_liegtin_AG
    WHERE AGNr_enthaelt=AG1)
  UNION ALL
  (SELECT f.AGNr_liegt_in,f.AGNr_enthaelt,i+1
```

```

FROM zwischenspeicher AS e, WWWT_AG_liegtin_AG AS f
  WHERE e.AGNr_liegt_in=f.AGNr_enthaelt
  AND i<25))
(SELECT DISTINCT AufgabenNr FROM WWWT_Au_liegtin_AG
  WHERE WWWT_Au_liegtin_AG.AGNr IN (SELECT AGNr_liegt_in
    FROM zwischenspeicher));

```

Die Variable i wird als Zähler eingeführt, um für die Rekursion auf jeden Fall ein Abbruchkriterium zu spezifizieren. Es ist möglich, daß beispielsweise AG1 in AG2 liegt, AG2 in AG3, und AG3 wiederum in AG1. Ohne einen Zähler würde die obige rekursive SQL-Anfrage in diesem Fall nicht abbrechen. Der Abbruch nach 25 Durchläufen ist frei gewählt, es könnte auch eine andere Zahl verwendet werden⁴.

Diese Anfrage benötigt aufgrund der Rekursion einen hohen Aufwand an Rechenzeit und Speicherplatz (im Vergleich zu einer nicht-rekursiven Anfrage auf derselben Relationen). Zusätzlich wird diese Anfrage (welche Aufgaben liegen in einem bestimmten Aufgabengebiet) häufig gestellt werden. Bei jeder Testgenerierung wird diese Anfrage für jedes verwendete Aufgabengebiet benötigt.

Um die Ausführungszeit dieser Anfrage zu optimieren wird deshalb in der Relation `WWWT_Au_liegtin_AG` explizit für jedes Aufgabengebiet AGx und jede darin enthaltene Aufgabe Ax ein Tupel (AGx,Ax) gespeichert. Durch diese Festlegung ergibt sich für die Frage nach allen Aufgaben für ein Aufgabengebiet eine nicht-rekursive SQL-Anfrage auf der Relation `WWWT_Au_liegtin_AG`. Es muß allerdings erwähnt werden, daß die Relation aufgrund obiger Festlegung mehr Tupel enthält.

Die genauen Methoden, um die eben genannte Festlegung sicherzustellen, werden im folgenden Abschnitt *Konsistenzerhaltung in der Datenbank* beschrieben. Prinzipiell muß beim Einfügen eines Tupels (AG1,AG2) in die Relation `WWWT_AG_liegtin_AG` folgendes beachtet werden: Für AG2 und alle AG2 enthaltene Aufgabengebiete muß in die Relation `WWWT_Au_liegtin_AG` für jede in AG1 liegende Aufgabe ein neues Tupel eingefügt werden (sofern noch nicht vorhanden).

Bei Eintragungen in `WWWT_Au_liegtin_AG` müssen zusätzlich alle Aufgabengebiete bestimmt werden, die dem Aufgabengebiet des eingefügten Tupels entsprechen.

Beim Löschen des Tupels (A1,AG1) in `WWWT_Au_liegtin_AG` müssen zusätzlich alle Tupel gelöscht werden, die A1 enthalten und deren Aufgabengebiet AGx in AG1 enthalten ist oder AG1 umfaßt oder ein Aufgabengebiet besitzen, das in AGx enthalten ist, usw. (die Beziehung wird rekursiv fortgesetzt)⁵.

Das DB-System muß andere Integritätsbedingungen gewährleisten, als wenn in `WWWT_Au_liegtin_AG` nicht alle Beziehungen zwischen Aufgaben und Aufgabengebieten gespeichert werden. Dies hat sowohl positive als auch negative Auswirkungen. Verdeutlicht

⁴Mir erscheint 25 eine angemessen große Zahl zu sein, so daß keine Beziehung bei dieser Anfrage übergangen wird.

⁵Zur Konsistenzerhaltung ist es nur erforderlich, A1 aus den Aufgabengebieten zu löschen, die AG1 enthält. Mir erscheint es jedoch sinnvoll, A1 auch aus den Aufgabengebieten zu löschen, in denen AG1 liegt, da A1 im Regelfall durch die Beziehung "AG1 liegt Aufgabengebiet" in das Aufgabengebiet gelangt ist.

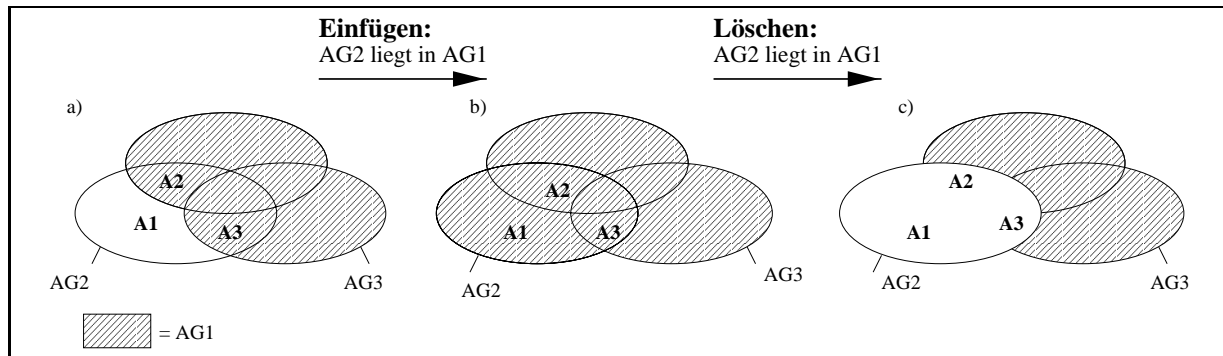


Abbildung 5.7: Verhalten der Datenbank beim Einfügen und Löschen von Beziehungen zwischen Aufgabengebieten

werden soll dies anhand eines Beispiels: Es existieren die Aufgabengebiete AG1, AG2 und AG3 wobei bereits AG3 in AG2 enthalten ist (siehe Abbildung 5.7a). Aufgabe A1 liegt in AG2, die Aufgabe A2 liegt in AG2 und AG1 (schraffierter Bereich), Aufgabe A3 liegt in AG1 und AG3. Durch das Einfügen der Beziehung "AG2 liegt in AG1" enthalten sowohl AG2 als auch AG1 die Aufgaben A1, A2 und A3 (siehe Abbildung 5.7b). AG3 enthält nach wie vor A3. Das Löschen der Beziehung "AG2 liegt in AG1" führt dazu, daß alle Aufgaben aus AG2 (A1, A2, A3) in AG1 gelöscht werden (siehe Abbildung 5.7c). Um A3 aus AG1 zu löschen, muß zusätzlich A3 aus AG3 entfernt werden. Dies folgt daraus, daß AG3 in AG1 enthalten ist. Würde A3 in AG3 aber nicht in AG1 enthalten sein, würde dies ein inkonsistenter Zustand innerhalb der Datenbank sein.

Das Einfügen und anschließende Löschen der Beziehung "AG2 liegt in AG1" führt also dazu, das in AG1 und sogar in AG3 weniger Aufgaben als vor diesen Operationen enthalten sind. Ob dies ein Vor- oder Nachteil ist, ist von der Semantik der Operation abhängig. Einerseits kann es erwünscht sein, daß beim Löschen der Beziehung "AG1 liegt in AG2" keine Aufgabe aus AG1 mehr in AG2 enthalten ist, andererseits ist es vorstellbar, trotz des Löschens der Beziehung AG1 liegt in AG2 einzelne Aufgaben in beiden Gebieten zu belassen. Dies ist durch die Festlegung, das alle Beziehungen zwischen Aufgaben und Aufgabengebieten in `WWW_T_Au_liegt_in_AG` steht, jedoch nicht mehr möglich⁶. Nach dem Löschen der Beziehung müssen, falls gewünscht, die Aufgaben wieder "von Hand" dem Aufgabengebiet AG2 zugeordnet werden.

Prinzipiell wäre es wünschenswert, bei der Operation "Lösche die Beziehung AG1 liegt in AG2" zwei Optionen zur Verfügung gestellt zu bekommen. Zum einen die oben vorgestellte Möglichkeit, in der alle Aufgaben, die in AG1 enthalten sind aus AG2 gelöscht werden. Zum anderen sollte es auch möglich sein, nur die durch die Beziehung "AG1 liegt in AG2" eingefügten Aufgaben aus AG2 zu löschen. Die zweite Option erfordert jedoch eine Erweiterung des DB-Schemas und wird in dieser Arbeit nicht weiter verfolgt.

⁶Durch die Erweiterung der Datenbank könnte auch diese Information gespeichert werden.

5.4 Konsistenzerhaltung in der Datenbank

Die Konsistenz der Daten in der Datenbank wird über verschiedene Mechanismen sichergestellt. Dazu zählen die Definition von *Primärschlüsseln* und *Fremdschlüsseln*, die Definition von *Check-Constraints* und die Definition von *Triggern* (siehe [Cha96]).

5.4.1 Primärschlüssel

Durch die Definition eines Primärschlüssels wird erreicht, daß in eine Relation kein zweites Tupel mit dem gleichen Primärschlüssel eingetragen werden kann. Beispielsweise kann in der Relation `WWWT_Aufgabe` jede `AufgabenNr` nur einmal verwendet werden. Damit besitzt jede Aufgabe einen eindeutigen Bezeichner und es kann nicht zu Mehrdeutigkeiten beim Einlesen einer Aufgabe kommen.

5.4.2 Fremdschlüssel

Fremdschlüssel garantieren, daß das als Fremdschlüssel definierte Attribut nur Werte annehmen kann, die in der entsprechenden Relation bereits vorhanden sind. Beispielsweise kann in der Relation `WWWT_Au_liegtin_AG` das Attribut `AufgabenNr` nur Werte annehmen, die bereits in der Relation `WWWT_Aufgabe` im Attribut `AufgabenNr` gespeichert sind. Damit wird erreicht, daß lediglich auf die Aufgaben in `WWWT_Au_liegtin_AG` Bezug genommen werden kann, die bereits vorher in `WWWT_Aufgabe` eingefügt wurden.

Ebenso wird durch Fremdschlüssel verhindert, daß Werte in einer Relation gelöscht werden, auf die sich eine andere Relation in einem Fremdschlüssel bezieht. So kann beispielsweise in der Relation `WWWT_Aufgabe` eine Aufgabe nur gelöscht werden, wenn in der Relation `WWWT_Au_liegtin_AG` kein Attributwert mehr auf diese Aufgabe verweist. Damit wird verhindert, das in `WWWT_Au_liegtin_AG` auf eine nicht mehr vorhandene Aufgabe verwiesen wird.

Die Fremdschlüssel der einzelnen Relationen der Datenbank sind bereits in Abschnitt 5.2 beschrieben.

5.4.3 Check-Constraints

Check-Constraints dienen dem Überprüfen von Konsistenzbedingungen beim Einfügen oder Verändern von Werten in der Datenbank. Check-Constraints können nur auf Attribute der einzufügenden oder zu verändernden Tupel zugreifen. Sie dienen in der Regel der Überprüfung, ob bestimmte Wertebereiche für einzelne Attribute eingehalten werden. Im folgenden wird ein Check-Constraint anhand eines Beispiels erläutert. Dazu wird zunächst der SQL-Code zur Erzeugung der Relation `WWWT_Item` dargestellt:

```
CREATE TABLE WWWT_Item (  
    AufgabenNr    INTEGER NOT NULL,  
    Position      INTEGER NOT NULL,
```

```

Text          VARCHAR(254) NOT NULL,
Korrektheit   CHAR(1) NOT NULL
CHECK (Korrektheit='T' OR Korrektheit='F' ),
PRIMARY KEY (AufgabenNr, Position),
FOREIGN KEY (AufgabenNr) REFERENCES WWWT_Aufgabe );

```

Mit der Check-Constraint (Korrektheit='T' OR Korrektheit='F') wird erreicht, daß das Attribut Korrektheit nur die Werte 'T' oder 'F' annehmen kann. Das Einfügen von anderen Werten wird vom DB-System mit einer Fehlermeldung verweigert. Ebenso ist die Veränderung des Attributwerts nicht möglich.

Die Check-Constraints der anderen Relationen der Datenbank ergeben sich aus den in Abschnitt 5.2 dargestellten Bedingungen und werden nicht im einzelnen vorgestellt.

5.4.4 Trigger

Trigger dienen ebenso wie Check-Constraints der Einhaltung bestimmter Konsistenzbedingungen. Trigger können im Gegensatz zu Check-Constraints auch auf andere Werte als die der einzufügenden oder zu verändernden Tupel innerhalb der Datenbank zugreifen. Damit stellen Trigger ein mächtigeres Werkzeug zur Definition von Konsistenzbedingungen dar. Trigger können sowohl für das Einfügen, als auch für das Verändern oder Löschen innerhalb einer Relation definiert werden.

Der folgende SQL-Code stellt die Definition für einen Trigger dar:

```

CREATE TRIGGER WWWT_AufgabeU NO CASCADE BEFORE
  UPDATE ON WWWT_Aufgabe
  REFERENCING NEW AS newRow
  FOR EACH ROW MODE db2sql
  WHEN (newRow.AufgabenNr IN (
      SELECT AufgabenNr FROM WWWT_Testprotokoll))
  SIGNAL SQLSTATE '70002'
  ('Aufgabe already in use (no more update possible)');

```

Mit diesem Trigger wird erreicht, daß eine Aufgabe nicht mehr verändert werden kann, sobald sie bereits während einer Testdurchführung verwendet wurde, also in WWWT_Testprotokoll referenziert wird. Die Notwendigkeit hierfür ergibt sich daraus, daß sich in WWWT_Testprotokoll und WWWT_Gew_Loesung auf eine bestimmte Aufgabe bezogen wird. Eine Veränderung dieser Aufgabe darf deshalb nicht mehr erfolgen⁷. Wird versucht, eine Aufgabe zu verändern, die der obigen Bedingung entspricht, bricht das DB-System die Operation mit einer Fehlermeldung ab.

Es folgt eine Beschreibung der für die Datenbank erzeugten Trigger, wobei lediglich die Funktion und nicht der SQL-Code beschrieben wird.

⁷Sollte sich bei der Aufgabenanalyse herausstellen, daß die Aufgabe schlecht gestellt wurde, muß die verbesserte Aufgabe unter einer neuen AufgabenNr gespeichert werden. Die vorhandene Aufgabe darf keinem Aufgabengebiet mehr zugewiesen werden; damit wird garantiert, daß die Aufgabe nicht mehr verwendet wird.

5.4.4.1 WWWT_AufgabeU

Der bereits eben beschriebene Trigger verhindert, das bereits verwendete Aufgaben in WWWT_Aufgabe verändert werden.

5.4.4.2 WWWT_Auf_AnzahlU

Mit diesem Trigger wird erreicht, daß ein Verändern der Punktzahl einer Aufgabe in WWWT_Aufgabe nur möglich ist, falls bei einer Verringerung der Punktzahl trotzdem für alle Tests eine genügend große Anzahl von Aufgaben mit der notwendigen Punktzahl existiert⁸.

5.4.4.3 WWWT_ein_ItemI und WWWT_ein_ItemU

Diese Trigger sorgen dafür, daß jede Multiple-Choice-Aufgabe ohne Mehrfachnennung maximal ein richtiges Item in der Relation WWWT_Item enthalten kann. Ein Trigger ist für das Einfügen, der andere für ein Verändern der Relation WWWT_Item definiert⁹.

5.4.4.4 WWWT_Item_TestTypI und WWWT_Item_TestTypU

Mit diesen beiden Triggern wird sichergestellt, daß in der Relation WWWT_Item nur Aufgaben vom Typ MCOMN und MCOMMN enthalten sein können. Dies kann nicht mit einem Check-Constraint erreicht werden, da die entsprechende Information (der Typ der Aufgabe) in einer anderen Relation steht.

5.4.4.5 WWWT_ItemU und WWWT_Item_neuI

Diese Trigger verhindern, daß in WWWT_Item bereits verwendete Items (AufgabenNr bereits in WWWT_Testprotokoll) verändert werden oder für bereits verwendete Aufgaben neue Items eingefügt werden.

5.4.4.6 WWWT_LW_Test_TypI und WWWT_LW_Test_TypU

Durch diese Trigger wird ausgeschlossen, daß in der Relation WWWT_Loesungswort Aufgaben enthalten sind, die nicht den Typ EGA besitzen.

⁸Beim Löschen einer Aufgabe muß dies nicht überprüft werden, da Aufgaben erst gelöscht werden können, wenn kein Fremdschlüssel einer anderen Relation auf sie verweist. D.h., daß die Aufgabe bereits in keinem Aufgabengebiet mehr liegen und damit auch in keinem Test mehr enthalten sein kann, bevor sie gelöscht werden darf.

⁹Jeder Trigger kann entweder für das Einfügen (insert), das Verändern (update) oder das Löschen (delete) definiert werden. Falls ein Trigger sowohl für das Einfügen als auch für das Verändern verwendet werden soll, müssen also zwei Trigger definiert werden. Trigger für das Einfügen haben in dieser Ausarbeitung im Namen am Schluß ein I (insert), Trigger zum Verändern ein U (update) und Trigger zum Löschen ein D (delete).

5.4.4.7 WWWT_LW_neuI und WWWT_LW_neuU

Sobald die von `WWWT_Loesungswort` referenzierte Aufgabe verwendet wurde, darf kein Verändern oder Einfügen neuer Lösungswörter für diese Aufgabe erfolgen.

5.4.4.8 WWWT_TestU

Es dürfen nur Tests in `WWWT_Test` verändert werden, die noch nicht verwendet wurden, also noch nicht in `WWWT_Testprotokoll` referenziert werden.

5.4.4.9 WWWT_Gen_TestI und WWWT_Gen_TestU

Mit diesen Triggern wird erreicht, daß in der Relation `WWWT_GenT_enth_AG` nur Tests vom Typ GEN enthalten sind.

5.4.4.10 WWWT_Gen_AnzahlI und WWWT_Gen_AnzahlU

Die in der Relation `WWWT_GenT_enth_AG` angegebenen Anzahl von Punkten muß mit den bereits vorhandenen Aufgaben generiert werden können.

5.4.4.11 WWWT_CAT_Test1I und WWWT_CAT_Test1U

Mit diesen Triggern wird erreicht, daß in der Relation `WWWT_CAT_enth_AG` nur Tests vom Typ CAT enthalten sind.

5.4.4.12 WWWT_CAT_Test_SSI und WWWT_CAT_Test_SSU

Diese Trigger sorgen dafür, daß in `WWWT_CAT_enth_AG` nur Tests und Aufgabengebiete enthalten sind, für die mindestens eine Schwierigkeitsstufe in `WWWT_CAT_kann_verw` steht.

5.4.4.13 WWWT_CAT_AnzahlI und WWWT_CAT_AnzahlU

Es wird sichergestellt, daß die in der Relation `WWWT_CAT_enth_AG` angegebenen Anzahl von Punkten mit den vorhandenen Aufgaben generiert werden kann.

5.4.4.14 WWWT_CAT_Test2I und WWWT_CAT_Test2U

Mit diesen Triggern wird erreicht, daß in der Relation `WWWT_CAT_kann_verw` nur Tests vom Typ CAT enthalten sind.

5.4.4.15 WWWT_SS_AnzahlI und WWWT_SS_AnzahlU

Diese beiden Trigger sorgen dafür, daß in `WWWT_CAT_kann_verw` nur Schwierigkeitsstufen eingetragen werden, für die in der Relation `WWWT_CAT_enth_AG` angegebene Punktzahl mit den in `WWWT_Aufgabe` enthaltenen Aufgaben erreicht werden kann.

5.4.4.16 WWWT_TestinstanzU

Eine Testinstanz aus `WWWT_Testinstanz` darf lediglich verändert werden, wenn die Testinstanz noch nicht bearbeitet wurde, also noch nicht in `WWWT_Testprotokoll` referenziert wird. Eine Ausnahme bildet die `BANr`, die auch anschließend verändert werden darf.

5.4.4.17 WWWT_Gew_fuer_SSU

Durch diesen Trigger wird verhindert, das eine Veränderung in `WWWT_Gew_fuer_SS` vorgenommen wird, falls die Gewichtung bereits verwendet wurde.

5.4.4.18 WWWT_Aufg_vorgabeI und WWWT_Aufg_vorgabeU

In der Relation `WWWT_Aufg_vorgabe` dürfen nur Tests vom Typ `GEN` enthalten sein.

5.4.4.19 WWWT_BewertungsAU

In `WWWT_Bewertungsart` darf keine Veränderung vorgenommen werden, wenn in `WWWT_Testinstanz` die Bewertungsart referenziert wird.

5.4.4.20 WWWT_NoteU

Eine Note aus `WWWT_Note` darf nicht mehr verändert werden, sobald sie in `WWWT_Notenschl` referenziert wird.

5.4.4.21 WWWT_NotenschIU

Der Trigger gewährleistet, das keine Veränderung in `WWWT_Notenschl` vorgenommen wird, sobald der Notenschlüssel (Über die `BANr`) in `WWWT_Testinstanz` referenziert wird.

5.4.4.22 WWWT_Au_keinUpdate

Mit diesem Trigger wird erreicht, daß keine Veränderung eines Tupels in der Relation `WWWT_Au_liegtin_AG` möglich ist. Sie dürfen lediglich neu eingefügt und gelöscht werden. Es wäre möglich ein Verändern dieser Relation zuzulassen. In diesem Fall müßte der Trigger die Operationen der Trigger zum Löschen (`WWWT_Au_loeschenD`, `WWWT_Au_AnzahlD`) mit dem alten Wert sowie den Trigger zum Einfügen (`WWWT_Au_erweitern`) mit dem neuen Wert durchführen.

Eine Veränderung von `WWWT_Au_liegtin_AG` wird jedoch ausgeschlossen, um den Benutzenden durch die dadurch erforderlichen Lösch- und Einfüge-Operationen die Tragweite dieses Handelns vor Auge zu führen¹⁰.

¹⁰Ich kann mir keinen Fall vorstellen, in dem eine Veränderung in der Relation sinnvoll erscheint. Wurde versehentlich ein falsche Beziehung eingefügt, erscheint mir der Weg, diese Beziehung zu Löschen und eine andere Einzufügen sinnvoller, als die unerwünschte Beziehung zwischen zwei Aufgabengebieten durch ein Verändern zu Löschen.

5.4.4.23 WWWT_AG_keinUpdate

In der Relation `WWWT_AG_liegtin_AG` darf keine Veränderung eines Tupels durchgeführt werden. Dies erfolgt aus dem gleichen Grund wie bei `WWWT_Au_keinUpdate`.

5.4.4.24 WWWT_AG_erweiternI

Mit diesem Trigger wird erreicht, daß beim Einfügen des Tupels (AG1,AG2) in der Relation `WWWT_AG_liegtin_AG` alle Aufgaben, die in AG1 liegen, auch in AG2 und in alle AG2 enthaltende Aufgabengebiete eingefügt wird. Dazu werden neue Tupel in die Relation `WWWT_Au_liegtin_AG` eingefügt.

Dieser Trigger trägt lediglich für jede Aufgabe, die in AG1 liegt, ein Tupel mit der entsprechenden Aufgabennummer und Aufgabengebiet in `WWWT_Au_liegtin_AG` ein (falls noch nicht vorhanden). Die Rekursion, d.h. das Auffinden aller Aufgabengebiete die indirekt das Aufgabengebiet AG1 enthalten, erfolgt über den entsprechenden Trigger `WWWT_Au_erweiternI` von `WWWT_Au_liegtin_AG`.

5.4.4.25 WWWT_Au_erweiternI

Es ist nötig, bei einem Eintrag in die Relation `WWWT_Au_liegtin_AG` zu überprüfen, ob das entsprechende Aufgabengebiet in einem anderen Aufgabengebiet enthalten ist. In diesem Fall muß für jedes dieser Aufgabengebiete ein weiterer Eintrag (falls noch nicht vorhanden) in `WWWT_Au_liegtin_AG` gemacht werden, der das übergeordnete Aufgabengebiet und die gleiche Aufgabe enthält (Begründung siehe 5.3). Innerhalb des Triggers muß keine rekursive Anfrage an `WWWT_AG_liegtin_AG` gestellt werden, da ein weiterer Eintrag in `WWWT_Au_liegtin_AG` einen rekursiven Aufruf dieses Triggers zur Folge hat. Die Rekursion endet auch, wenn die Aufgabengebiete eine Schleife enthalten (z.B. AG1 liegt in AG2; AG2 liegt in AG3 und AG3 in AG1), da nur noch nicht vorhandene Werte eingetragen werden.

5.4.4.26 WWWT_AG_loeschenD

Dieser Trigger löscht die durch die Trigger `WWWT_AG_erweiternI` und `WWWT_Au_erweiternI` in der Relation `WWWT_Au_liegtin_AG` eingetragenen Tupel.

Dabei kann es vorkommen, daß Aufgaben, die vorher in den Aufgabengebieten AG1 und AG2 gelegen sind, durch daß Einfügen und anschließende Löschen des Tupels (AG1,AG2) in der Relation `WWWT_AG_liegtin_AG`, nicht mehr in AG2 liegen, da der Trigger `WWWT_AG_loeschenD` diese Beziehung gelöscht hat (siehe auch 5.3).

5.4.4.27 WWWT_Au_loeschenD

Mit diesem Trigger wird erreicht, das beim Löschen eines Tupels (Au1,AG1) in der Relation `WWWT_Au_liegtin_AG` alle Tupel gelöscht werden, die Au1 als AufgabenNr enthalten und deren Aufgabengebiet entweder in AG1 enthalten ist oder AG1 enthält. Da jede Aufgabe in `WWWT_Au_liegtin_AG` explizit dem Aufgabengebiet zugeordnet wird, würde das Löschen

eines Tupels ohne die Berücksichtigung der Aufgabengebiete, die in dem Aufgabengebiet liegen oder das Aufgabengebiet enthalten, zu einem inkonsistenten Zustand führen. Die Rekursion zur Bestimmung aller betroffenen Aufgabengebiete erfolgt nicht in einer SQL-Anfrage innerhalb des Triggers, sondern ergibt sich durch den erneuten Aufruf des Triggers beim Löschen der betroffenen Tupel.

5.4.4.28 WWWT_Au_AnzahlD

Dieser Trigger verhindert, daß in `WWWT_Au_liegtin_AG` Aufgaben aus Aufgabengebieten gelöscht werden, falls die benötigten Punkte eines Tests, der das Aufgabengebiet enthält, sonst nicht mehr erreicht werden kann.

Kapitel 6

Entwurf der Werkzeuge

In diesem Kapitel wird der Entwurf der Werkzeuge zur Testdurchführung sowie zur Auswertung von Tests beschrieben. Das in den Anforderungen (siehe Kapitel 4) beschriebene Werkzeug zur Test- und Aufgabenadministration wird im Rahmen des Entwurfs nicht weiter betrachtet, da die wesentlichen Anforderungen bereits innerhalb des Datenbankentwurfs (siehe Abschnitt 5.4) berücksichtigt wurden.

6.1 Testdurchführung

Das Werkzeug zur Testdurchführung (kurz WETED) dient sowohl der Bearbeitung von Tests als auch der Bereitstellung der erzielten Rohwerte, der richtigen Lösungen der verwendeten Aufgaben sowie, falls gewünscht, der erzielten Noten. Zuerst wird der zeitliche Ablauf bei der Durchführung eines Tests beschrieben. Anschließend wird die Oberfläche des WETED skizziert und auf die Generierung der Testfragen sowie dem Verhalten im Fehlerfall eingegangen.

6.1.1 Zeitlicher Ablauf

6.1.1.1 Aufruf des Testprogramms

Eine Teilnehmende, die einen Test durchführen will, muß zunächst mit ihrem WWW-Browser die entsprechende HTML-Seite aufrufen, auf der sich das WETED-Applet befindet. Für jede Testinstanz muß eine eigene HTML-Seite zur Verfügung stehen. In der HTML-Seite werden dem Applet bereits die Parameter übergeben, um welche Testinstanz es sich bei der Ausführung handelt. Prinzipiell wäre es auch möglich, für alle Testinstanzen eines Tests oder sogar alle Tests nur eine HTML-Seite zur Verfügung zu stellen. In diesem Fall müßte die Teilnehmende jedoch die entsprechende Testinstanz selbst auswählen. Dies kann dazu führen, daß die Teilnehmende die falsche Testinstanz wählt und bearbeitet.

6.1.1.2 Authentifizierung

Beim Start wird als erstes Name des Teilnehmenden¹ sowie das Passwort des Teilnehmenden abgefragt. Falls der Teilnehmende die Testinstanz nicht bearbeiten darf (kein Eintrag in `WWWT_Teilnahme`) oder ein falsches Passwort eingegeben wurde, wird eine entsprechende Meldung angezeigt und anschließend erneut die Eingabe des Namens des Teilnehmenden und des Passwort verlangt. Wurde die Testinstanz vom Teilnehmenden bereits vollständig bearbeitet, wird ihm die Möglichkeit zur Einsicht gegeben (siehe Abschnitt 6.1.1.5). Falls die Bearbeitung der Testinstanz durch den Teilnehmenden noch erfolgen muß, wird, sofern dies für die Testinstanz erforderlich ist, die Eingabe des Passworts der Aufsicht verlangt. Damit die Authentifizierung des Teilnehmenden durch die Aufsicht gewährleistet werden kann, werden der Aufsicht die Daten aus der Relation `WWWT_Teilnahme` angezeigt. Nach dem vorliegenden Entwurf der Datenbank sind dies die Attribute `VName`, `NName` sowie `MatrNr`. Durch eine Erweiterung des Datenbankschemas wäre es vorstellbar, ein Bild des Teilnehmenden anzuzeigen. Wurde von der Aufsicht ein falsches Passwort eingegeben, werden nach einer entsprechenden Fehlermeldung erneut Name des Teilnehmenden sowie Passwort verlangt.

6.1.1.3 Anleitung

Nach der erfolgreichen Authentifizierung des Teilnehmenden wird zunächst die Testanleitung angezeigt. Sobald die Testanleitung angezeigt wird kann das Programm bereits mit der Generierung der Aufgaben sowie dem Einlesen der generierten Aufgaben beginnen. Die Anleitung wird solange angezeigt, bis der Teilnehmende sie bestätigt.

6.1.1.4 Bearbeitung des Tests

Nach der Bestätigung der Anleitung beginnt die Bearbeitung des Tests. Damit einem Teilnehmenden nicht Bearbeitungszeit verloren geht, werden zunächst, soweit möglich, alle Aufgaben geladen. Erst anschließend beginnt die Zeitnahme. Je nach Ausprägung der Testinstanz unterscheiden sich die Möglichkeiten bei der Bearbeitung einer Testinstanz. Bei einem Test mit Aufgabengenerierung kann es möglich sein, zwischen den einzelnen Aufgaben zu wechseln. Andernfalls muß jede angezeigte Aufgabe sofort bearbeitet werden. Je nach Ausprägung werden die richtigen Lösungen der Aufgaben sofort nach der Bearbeitung der Aufgabe angezeigt oder gleich die nächste Aufgabe gestellt.

Die Bearbeitung endet, wenn alle Aufgaben bearbeitet wurden, die Bearbeitungszeit abgelaufen ist oder der Teilnehmende die Bearbeitung vorzeitig beendet. Im letzten Fall werden alle noch nicht bearbeiteten Aufgaben als unbearbeitet gewertet. Nach der Bearbeitung des Tests wird sofort der erzielte Rohwert berechnet. Dieser wird in der Datenbank abgelegt.

¹Da der Primärschlüssel der Relation `WWWT_Teilnehmende` die `TNr` ist, wird unter dem Namen des Teilnehmenden in diesem Fall der Attributwert von `TNr` verstanden.

6.1.1.5 Einsichtnahme der Ergebnisse

Nach der Beendigung des Tests werden dem Teilnehmenden sofort die bereits vorliegenden Ergebnisse angezeigt. Zusätzlich kann der Teilnehmende durch ein erneutes Starten des Programms mit der gleichen Testinstanz alle zu diesem Zeitpunkt vorliegenden Ergebnisse angezeigt bekommen.

Die Ergebnisse sind im einzelnen der erreichte Rohwert, der maximal erreichbare Rohwert sowie die Bearbeitungszeit. Zusätzlich wird, falls gewünscht, die Note angezeigt. Falls zu diesem Zeitpunkt bereits erlaubt, können auf Wunsch zusätzlich die richtigen Lösungen, sowie die vom Teilnehmenden gewählten Lösungen der Aufgaben angezeigt werden.

6.1.2 Rohwertkorrektur

Der Zeitpunkt der Rohwertkorrektur kann einerseits für jede Aufgabe sofort nach deren Bearbeitung bei der Vergabe der Punktzahl erfolgen. Andererseits kann die Rohwertkorrektur bei der Ermittlung des Rohwerts nach der Bearbeitung des Tests erfolgen. Im zweiten Fall ergibt das Zusammenzählen der Punkte aller Aufgaben in der Regel einen anderen Wert als der bereits ermittelte Rohwert nach der Bearbeitung. Um Mißverständnisse bei der Einsichtnahme zu vermeiden, erfolgt die Rohwertkorrektur deshalb bereits bei der Vergabe der Punkte für die Aufgaben.

Die Ermittlung des Rohwerts unterscheidet sich nach Aufgabentyp und den für den Test festgelegten Auswertungskriterien.

Für *Ergänzungsaufgaben* findet keine Rohwertkorrektur statt. Die Art der Rohwertkorrektur von *Multiple-Choice-Aufgaben mit Mehrfachnennung*, wird mit den Attributen `Teilpunkte_erlaubt` und `Neg_Punktwerte` aus `WWWT_Test` festgelegt. Falls keine negativen Punktwerte erlaubt sind, wird als niedrigster erreichter Punktwert Null verwendet; keine Berücksichtigung von teilweise richtigen Antworten führt dazu, daß richtige Lösungen die volle Punktzahl erhalten und alle anderen Antworten mit Null Punkten bewertet werden.

Für *Multiple-Choice-Aufgaben ohne Mehrfachnennung* ist die Rohwertkorrekturart im Attribut `Rohwertkorrektur` aus `WWWT_Test` festgelegt. Falls als Rohwertkorrektur `AP` (Abzug der Punkte) festgelegt wurde, wird bei einer falschen Antwort die negative Punktzahl eingetragen. Falls keine Rohwertkorrektur festgelegt wurde, wird als Punktwert für falsch beantwortete Aufgaben Null verwendet. Falls als Rohwertkorrektur `ATP` (Abzug von Teilpunkten) gewählt wurde, wird für falsche Antworten die zu vergebende Punktzahl dividiert durch die Anzahl der Items als negativer Wert verwendet. Eine Zuzählung von Teilpunkten (`ZTP`) bewirkt, das falsch beantwortete Aufgaben mit Null Punkten, und unbeantwortete Aufgaben mit der zu vergebenden Punktzahl durch die Anzahl der Items bewertet werden.

6.1.3 Oberfläche zur Testdurchführung

Im folgenden wird die Oberfläche des WETED bei der Bearbeitung der Aufgaben beschrieben.

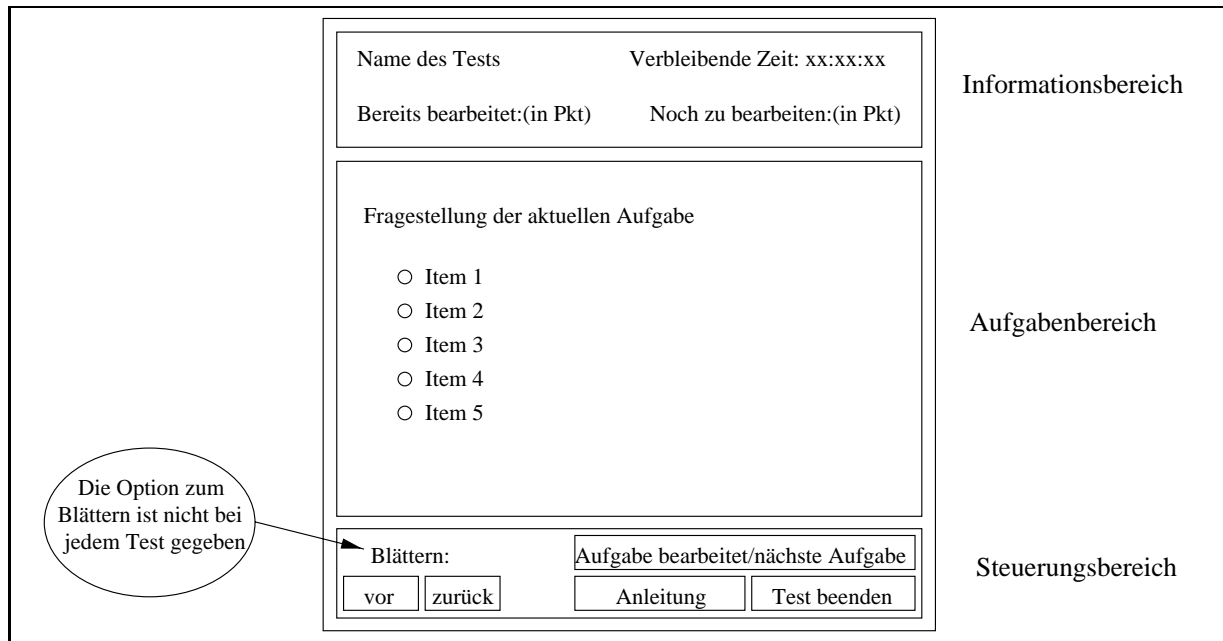


Abbildung 6.1: Die Oberfläche zur Testdurchführung im WETED

Die Oberfläche teilt sich in drei Bereiche. Ein Bereich zeigt Informationen an, ein zweiter die aktuell zu bearbeitende Aufgabe und ein dritter stellt die benötigten Steuerungselemente zur Verfügung (siehe Abbildung 6.1).

6.1.3.1 Informationsbereich

Im Informationsbereich wird der Testname, die verbleibende Zeit sowie die bereits bearbeiteten Aufgaben und die verbleibenden Aufgaben (über die Punktzahl) angezeigt.

6.1.3.2 Aufgabenbereich

Im Aufgabenbereich wird die zu bearbeitende Aufgabe angezeigt. Bei *Ergänzungsaufgaben* wird für jede Lücke ein Testfeld angezeigt, in das nach Anwahl durch die Maus das Lösungswort per Tastatur eingegeben werden kann.

Bei *Multiple-Choice-Aufgaben ohne Mehrfachnennung* stehen vor den Items Radio-Buttons, die mit der Maus angewählt werden können. Es kann lediglich ein Radio-Button gleichzeitig ausgewählt sein.

Bei *Multiple-Choice-Aufgaben mit Mehrfachnennung* stehen vor den Items Checkbox-Buttons, die mit der Maus angewählt werden können. Bei Checkbox-Buttons können mehrere Buttons gleichzeitig aktiviert werden.

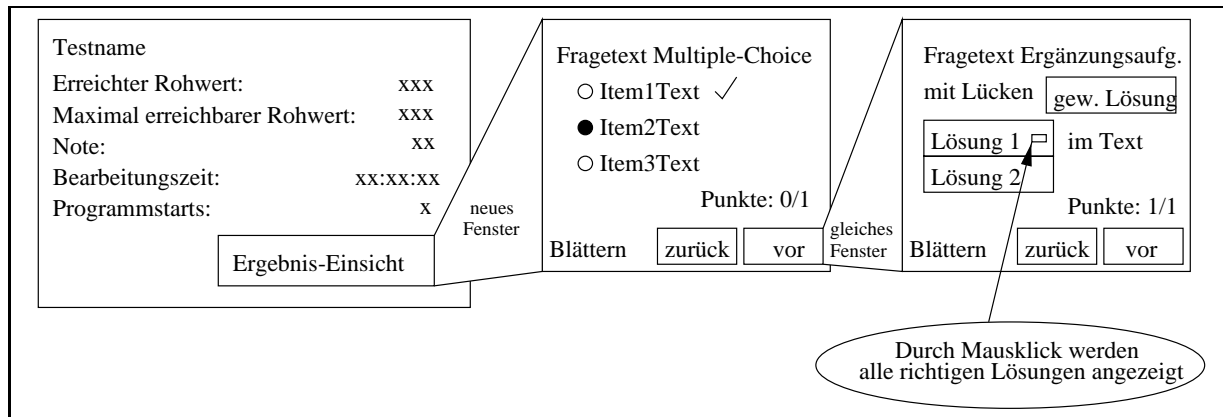


Abbildung 6.2: Die Oberfläche zur Einsicht im WETED

6.1.3.3 Steuerungsbereich

Im Steuerungsbereich werden immer die Buttons “Test beenden”, “Aufgabe bearbeitet/nächste Aufgabe” und “Anleitung” angezeigt sowie, falls für den Test vorgesehen, die Buttons “vor” und “zurück” für das Wechseln innerhalb der noch nicht beantworteten Aufgaben.

Das Betätigen von “Test beenden” bewirkt das Anzeigen einer Meldung, daß der Test beendet wird. Die Meldung zeigt zusätzlich die Anzahl der unbearbeiteten Aufgaben. Falls die Meldung bestätigt wird, wird die Bearbeitung des Tests vorzeitig beendet.

Mit “Aufgabe bearbeitet/nächste Aufgabe” wird erreicht, daß das Ergebnis der angezeigten Aufgabe gespeichert wird (auch wenn die Aufgabe unbeantwortet ist) und die nächste Aufgabe angezeigt wird, bzw. falls alle Aufgaben bearbeitet wurden der Test beendet wird. Falls als Freigabe der richtigen Lösungen SOF (Sofortige Freigabe) in `WWWT_Testinstanz` angegeben wurde, wird in einem neuer Fenster die richtige Lösung der Aufgabe angezeigt. Die Betätigung von “Anleitung” bewirkt, daß in einem weiteren Fenster die Testanleitung angezeigt wird. Ein weiterer Aufruf von “Anleitung” bewirkt, daß das Fenster wieder geschlossen wird.

6.1.4 Oberfläche zur Einsicht

Die Oberfläche zur Einsicht wird nach Beendigung einer Testinstanz sowie beim erneuten Starten einer bereits bearbeiteten Testinstanz angezeigt. Sie zeigt den erreichten Rohwert sowie den maximal erreichbaren Rohwert an. Zusätzlich wird die Bearbeitungszeit sowie die Anzahl der Programmstarts während der Bearbeitung² angezeigt. Von dieser Oberfläche aus kann, falls erlaubt, eine Einsicht in die richtigen Lösungen vorgenommen werden.

²In der Regel eins, falls das Programm während der Bearbeitung ausgefallen ist auch größere Werte.

6.1.4.1 Einsicht in die richtigen Lösungen

Die Einsicht der richtigen Lösungen erfolgt in einem neuen Fenster. In diesem Fenster wird genau eine Aufgabe angezeigt. Durch Blättern kann zwischen den Aufgaben gewechselt werden. Es wird bei jeder Aufgabe die vom Teilnehmenden gewählte sowie die richtige Lösung angezeigt. Falls bei Ergänzungsaufgaben verschiedene richtige Lösungen zugelassen sind, wird eine sofort und die Anderen nach der Anwahl des Lösungsfeldes angezeigt (siehe Abbildung 6.2). Für jede Aufgabe wird die erreichte sowie die mögliche Punktzahl angezeigt.

6.1.5 Testfragengenerator

Der Testfragengenerator dient der Auswahl der in einem Test verwendeten Aufgaben. Die Funktion des Testfragengenerators unterscheidet sich aufgrund der verschiedenen Testtypen.

Bei einem *Test mit Aufgabengenerierung*, bei dem alle Teilnehmenden die gleichen Aufgaben bearbeiten, muß zunächst in `WWWT_Aufg_vorgabe` überprüft werden, ob bereits Aufgaben für die Testinstanz generiert wurden. Falls bereits Aufgaben generiert wurden, müssen diese verwendet werden, ansonsten müssen die Aufgaben neu generiert werden und zunächst in `WWWT_Aufg_vorgabe` eingefügt werden.

Die Generierung erfolgt für jedes Aufgabengebiet der darin verwendeten Schwierigkeitsstufe. In der Relation `WWWT_GenT_ent_AG` steht, wieviel Aufgaben (indirekt über die Punktzahl) für eine Schwierigkeitsstufe eines Aufgabengebiets gestellt werden sollen. Mit der entsprechenden SQL-Anfrage (siehe Abschnitt 7.1.1) werden zunächst alle möglichen Aufgaben (liegen im Aufgabengebiet und besitzen die Schwierigkeitsstufe) ausgewählt. Es werden keine Ergebnisse vorheriger Bearbeitungen berücksichtigt.

Aus der Liste der verwendbaren Aufgaben werden per Zufall solange Aufgaben ausgesucht, bis entweder die vorgegebene Punktzahl erreicht wird (damit endet die Auswahl), oder durch die letzte ausgewählte Aufgabe *A1* die Punktzahl überschritten wird. Falls durch *A1* die geforderte Punktzahl überschritten wird, wird zuerst versucht mit einer noch nicht ausgewählten Aufgabe (oder mehreren Aufgaben) statt *A1* die geforderte Punktzahl zu erreichen. Scheitert auch dies, wird versucht mit *A1* und durch das Weglassen bereits ausgewählter Aufgaben eine möglichst gute Näherung (von oben) an die vorgegebene Punktzahl zu erreichen. Wird die vorgegebene Punktzahl noch überschritten, wird für die Aufgabe, mit der am wenigsten Punkte erzielt werden können, die Punktzahl herabgesetzt (nur für diese Aufgabengenerierung). Der Fall, in dem die Punktzahl herabgesetzt wird, wird in der Regel nicht auftreten, da für jede Aufgabe ein Punkt vergeben werden sollte.

Die eben beschriebene Methode gewährleistet nicht, daß alle Kombinationen ausprobiert werden. Deshalb ist es möglich, daß die Absenkung der Punktzahl für eine Aufgabe mit einer anderen Aufgabenkombination vermeidbar wäre. Die Vorteile dieser Methode liegen in der Geschwindigkeit (im Vergleich zum Ausprobieren aller Fälle) und darin, daß die Auswahl zu einem hohen Grad vom Zufall abhängt. Andere Methoden würden in einem höheren Grad bestimmte Aufgaben aufgrund ihrer Punktevergabe bevorzugen.

Bei *CAT-Tests* und *Tests mit Aufgabengenerierung*, bei dem für jeden Teilnehmenden eigene Aufgaben generiert werden erfolgt die Generierung ähnlich. Im Testmodus wird jedoch zuerst versucht, nur aus vom Teilnehmenden noch nicht bearbeiteten Aufgaben den Test zu generieren. Erst wenn nicht genügend unbearbeitete Aufgaben vorhanden sind, werden bereits bearbeitete verwendet.

Im Übungsmodus wird versucht, vorher falsch bearbeitete Aufgaben bevorzugt zu verwenden. Dazu wird für jede verwendbare Aufgabe eine Gewichtung berechnet. Höher gewichtete Aufgaben werden bevorzugt bei der Generierung verwendet.

Zur Berechnung der Gewichtung einer Aufgabe wird folgendes Verfahren verwendet: Es wird für die letzten fünf Lösungsversuche der Aufgabe durch den Teilnehmenden das Verhältnis der erreichten Punkte zu den maximal erreichbaren Punkten ermittelt. Falls keine fünf Lösungsversuche erfolgten, wird für die fehlenden als Wert 0,5 verwendet. Um die Reihenfolge der Lösungsversuche zu berücksichtigen wird der Wert des zuletzt gemachten Lösungsversuches mit 0,3 multipliziert, die vorherigen in zeitlicher Reihenfolge mit 0,25; 0,2; 0,15 bzw 0,1 für den am weitesten zurückliegenden Versuch. Diese fünf Werte werden addiert und ergeben so die Gewichtung. Sie kann einen Wert zwischen 1,0 und 0,0 annehmen.

6.1.6 Verhalten im Fehlerfall

Im folgenden wird kurz auf das Verhalten des Werkzeugs im Fehlerfall eingegangen.

6.1.6.1 Ausfall und Neustart

Falls das Werkzeug während der Bearbeitung eines Tests ausfällt, sollte es erneut gestartet werden. Die bereits bearbeiteten Aufgaben sind in der Datenbank persistent gespeichert und werden bei der Weiterbearbeitung berücksichtigt. Die Testzeit ist bis zum Zeitpunkt der letzten bearbeiteten Aufgabe ebenfalls in der Datenbank gespeichert und wird bei einem Neustart berücksichtigt. Die Anzahl der Neustarts wird ebenfalls in der Datenbank gespeichert, so daß bei bewerteten Tests gegebenenfalls aufgrund zu häufiger Ausfälle ein Testergebnis annulliert werden kann.

6.1.6.2 Inkonsistente Daten in Datenbank

Innerhalb der Konsistenzbedingungen der Datenbank wird nicht überprüft, ob jede Multiple-Choice-Aufgabe mindestens ein bzw. zwei Items besitzt, ebenfalls wie nicht die Anzahl der Lösungswörter für Ergänzungsaufgaben überprüft wird. Es ist also möglich, daß diese Daten in der Datenbank falsch bzw. nicht gespeichert sind³. Bei einer Multiple-Choice-Aufgabe mit einer ungenügenden Anzahl von Items wird eine Fehlermeldung ausgegeben und die Aufgabe als richtig gelöst bewertet. Falls bei Ergänzungsaufgaben für einige Lücken keine Lösungswörter spezifiziert wurden, wird ebenfalls eine Fehlermeldung ausgegeben

³Dies sollte durch ein Werkzeug zur Test- und Aufgabenadministration vermieden werden.

und die Lösungen als richtig gewertet. Lösungswörter einer Ergänzungsaufgabe, die keiner Lücke zugewiesen werden können, werden ignoriert. Unkorrekt gestellte Aufgaben als richtig zu bewerten erscheint in einem Test der fairste Weg zu sein. Vor allem bei bewerteten Tests sollte allerdings auf eine korrekte Aufgabenstellung geachtet werden und diese eventuell in einem Probedurchlauf getestet werden.

6.2 Auswertung

Das Werkzeug zur Auswertung dient dem Bewerten der Testergebnisse (Benotung) sowie der Analyse der Aufgaben und der Tests. Es wird im folgenden WEBEWAU (WERkzeug zur BEWertung und AUswertung) genannt. Es wird deshalb in die Bereiche *Bewertung der Ergebnisse*, *Aufgabenanalyse* sowie *Testanalyse* eingeteilt.

Zuerst wird der eingeschränkte Zugriff auf WEBEWAU beschrieben und der Ablauf beim Aufruf erläutert. Anschließend wird auf die einzelnen Bereiche des WEBEWAU eingegangen.

6.2.1 Zugriff auf WEBEWAU

WEBEWAU darf nur von autorisierten Personen benutzt werden. Um den unbefugten Zugriff zu vermeiden, darf es deshalb nicht frei zugänglich sein⁴. Prinzipiell ist der Schutz des verwendeten Betriebssystems ausreichend, um einen unerlaubten Zugriff zu verhindern. Es wäre allerdings vorstellbar, zusätzlich mit dem Abfragen einer Kennung sowie eines Passworts einen unerlaubten Zugriff zu verhindern. Dazu sollte eine zusätzliche Relation in der Datenbank verwendet werden, die diese Daten verwaltet⁵. Ein Vorteil wäre, daß neben der Zugangsberechtigung noch andere Informationen gespeichert werden können, die den Zugang limitieren. So kann es erforderlich sein, daß einige Personen lediglich die Aufgaben analysieren und keine Bewertung der Ergebnisse vornehmen dürfen. Dies kann in der Datenbank ebenfalls mitverwaltet werden.

6.2.2 Aufruf von WEBEWAU

Nach dem Start von WEBEWAU ist es erforderlich, einen der drei Bereiche (Bewertung, Aufgaben- oder Testanalyse) auszuwählen. Aufgrund der Auswahl wird ein Menü angezeigt, aus dem die entsprechenden Testinstanzen bzw. Aufgaben gewählt werden können. Zur Auswahl der Testinstanzen werden zuerst die `TestNr` und der entsprechende `TName` der möglichen Tests angezeigt und nach der Auswahl eines Tests die möglichen `TINr`. Die Auswahl der Aufgaben kann nach verschiedenen Kriterien erfolgen (siehe 6.2.4).

⁴Im Gegensatz zum WETED, bei dem ein freier Zugang erwünscht ist.

⁵Eine solche Relation ist nicht im Datenbankentwurf dieser Arbeit enthalten.

The screenshot shows a web-based evaluation interface. It is divided into several sections:

- Test Instance Information:** Fields for Testname, Instanz: TINr, Start Datum, End Datum, Start Zeit, End Zeit, and Anzahl Teilnehmende (33).
- Evaluation Type Selection:** Radio buttons for:
 - Normorientiert-Testinstanz als Norm
 - Normorientiert-Alle Testinstanzen als Norm
 - Kriteriumsorientiert
- Number of Grades:** A text input field containing the number 5.
- Grade Key Selection:** A dropdown menu showing 'Auswahl Notenschlüssel : BANr'. A callout box shows a list of available keys:
 - BANr: 1; Note 1:10%; Note 2 :20% :Note 3: 40
 - BANr: 5; Note 1:15%; Note 2 :25% : Note 3 25
 - BANr: 7; Note sehr gut:15%; Note gut :15% :No
 - BANr: 9; Note 1:15% ; Note 2 : 30% : Note 3 30
 - BANr: 12; Note sehr gut:10%; Note gut : 15% :N
 - BANr: 13; Note sehr gut:10%; Note gut : 20% :N
 - BANr: 17; Note sehr gut: 7%; Note gut :20% : N
- Action Buttons:** 'Übernehmen', 'Abbrechen', and 'Neuen Notenschlüssel eingeben'.
- Test Analysis Section:** Buttons for 'Testanalyse', 'Ergebnis betrachten', 'Notenschlüssel speichern (nicht anwenden)', 'Benotung anwenden', and 'Abbrechen'. A callout box shows the results of a test analysis:
 - Notenschlüssel: BANr
 - Durchschnitt: 2,7
 - Notenverteilung:

1	3
2	10
3	15
4	4
5	1

Abbildung 6.3: Die Oberfläche zur Bewertung der Ergebnisse im WEBEWAU

6.2.3 Bewertung der Ergebnisse

Die Bewertung der Ergebnisse erfolgt für jeweils eine Testinstanz. Es können lediglich die Testinstanzen im Testmodus bewertet werden, bei denen entweder alle Teilnehmenden den Test beendeten, oder der Bearbeitungszeitraum verstrichen ist (**End_Datum** und **End_Zeit**). Zusätzlich darf noch keine Bewertung vorgenommen worden sein. Eine bereits erteilte Bewertung darf nicht mehr zurückgenommen werden, da die Noten in diesem Fall bereits veröffentlicht wurden.

In Abbildung 6.3 ist die Oberfläche für die Bewertung der Ergebnisse dargestellt. Es werden Daten der Testinstanz angezeigt und es kann eine Bewertungsart, sowie die Anzahl der Noten gewählt werden. Falls bereits eine **BANr** für die Testinstanz gespeichert wurde, werden diese Werte als Default-Werte verwendet. Aufgrund der Bewertungsart und der Anzahl der Noten kann ein Notenschlüssel aus einer Liste gewählt werden. Es ist ebenso möglich, neue Notenschlüssel einzugeben⁶.

Die Testanalyse für die gewählte Testinstanz kann von dieser Oberfläche gestartet werden. Ebenso kann das Ergebnis angezeigt werden, daß sich aufgrund der gewählten Bewertungskriterien ergeben würde. Es wird der Notendurchschnitt sowie die Häufigkeit jeder Note angezeigt (in absoluten Zahlen). Bei dieser Darstellung wird absichtlich vermieden, die Noten einzelnen Teilnehmenden zuzuordnen, um zu vermeiden, daß einzelne Teilnehmende bevorzugt oder benachteiligt werden. Dies entspräche nicht einer objektiven Bewertung. Bei einer normorientierten Bewertung, bei der alle Testinstanzen verwendet werden, sollte

⁶Dies ist Aufgabe des Werkzeugs zur Test- und Aufgabenadministration und wird nicht weiter behandelt.

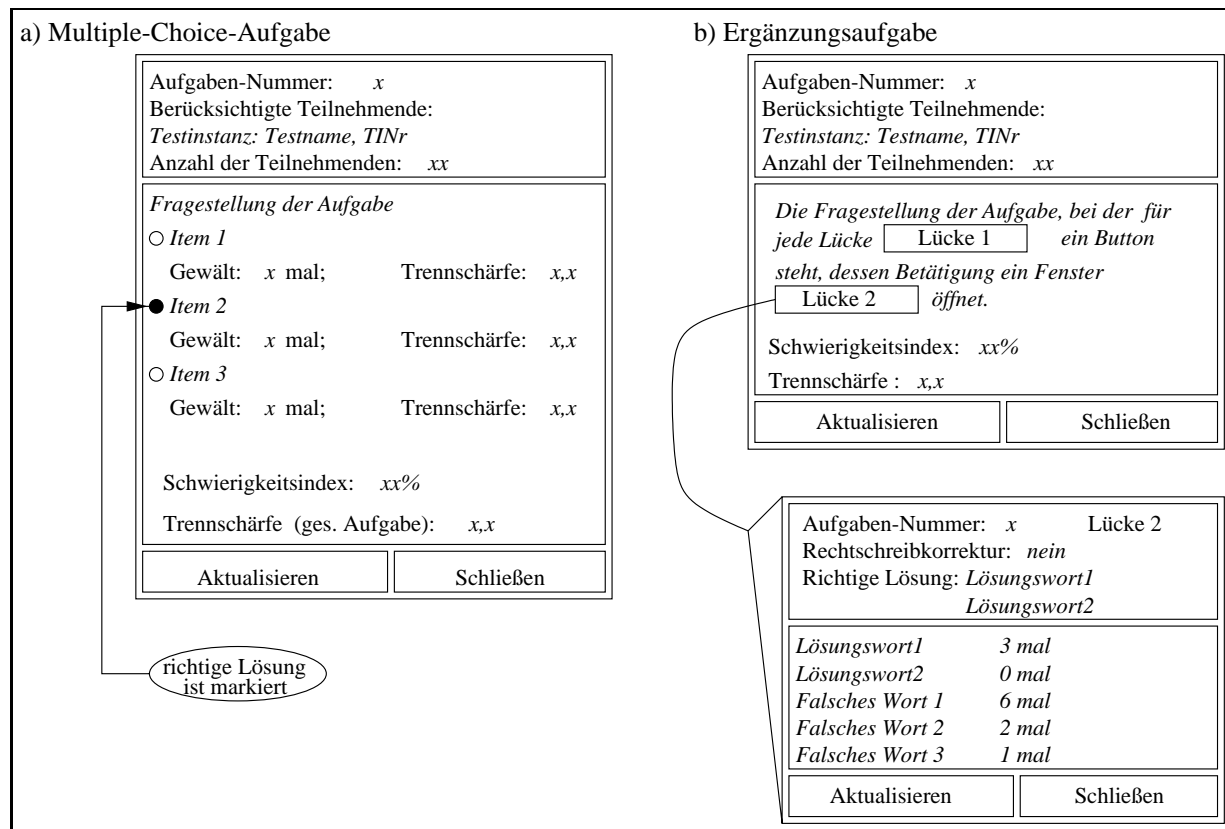


Abbildung 6.4: Die Oberfläche zur Aufgabenanalyse im WEBEWAU

neben dem Durchschnitt und der Notenverteilung der Testinstanz der Durchschnitt und die Notenverteilung aller Testinstanzen angezeigt werden. Der Button *Aktualisieren* wird benötigt, um bei einer erneuten Auswahl eines Notenschlüssels die angezeigten Werte anzupassen. Es ist möglich, mehrere Fenster mit den Ergebnissen der Bewertung zu öffnen, um die Ergebnisse vergleichen zu können. Dazu ist es erforderlich, daß die Ergebnisse nicht automatisch bei der Wahl eines Notenschlüssels aktualisiert werden.

Sobald *Bewertung anwenden* gewählt wurde (und ein entsprechendes Meldungsfenster angezeigt wurde) werden die Noten in der Datenbank gespeichert und nach einer Erfolgsmeldung das zugeordnete Fenster geschlossen.

6.2.4 Aufgabenanalyse

Die Auswahl der Aufgabe kann über das Aufgabengebiet, den Test oder die Testinstanz erfolgen. Ebenso sollte eine Aufgabe direkt über die *AufgabenNr* oder einen Suchstring innerhalb der *Fragestellung* gewählt werden können. Die Aufrund der Suchkriterien gefundenen Aufgaben werden in einer Liste angezeigt (*AufgabenNr* und die ersten Zeichen von *Fragestellung*). Aus dieser Liste kann eine Aufgabe gewählt werden, die daraufhin analysiert wird. Vor der Analyse muß angegeben werden, auf welche Teilnehmenden sich

Rangordnung			Prozentrangplatzierung		Standardwert	
Test: <i>Testname</i>			Test: <i>Testname</i>		Test: <i>Testname</i>	
Testinstanz: 23 Anzahl: 5			Testinstanz: 23 Anzahl: 5		Testinstanz: 23 Anzahl: 5	
Rang	Name, Vorname	Rohwert	Name, Vorname	Prozentrang	Name, Vorname	Standardwert
1	<i>Muster, Gerd</i>	16	<i>Muster, Gerd</i>	100%	<i>Muster, Gerd</i>	1,73
2	<i>Maier, Ines</i>	13	<i>Maier, Ines</i>	80%	<i>Maier, Ines</i>	1,23
3	<i>Nett, Rita</i>	10	<i>Nett, Rita</i>	60%	<i>Nett, Rita</i>	0,52
4	<i>Mann, Erna</i>	6	<i>Mann, Erna</i>	40%	<i>Mann, Erna</i>	-0,28
5	<i>Adler, Herbert</i>	2	<i>Adler, Herbert</i>	20%	<i>Adler, Herbert</i>	-1,09
Schließen			Schließen		Schließen	

Abbildung 6.5: Die Oberfläche der Testanalyse im WEBEWAU

die Analyse der Aufgabe beschränkt. Möglich sind dabei alle Teilnehmenden, die die Aufgabe bearbeitet haben (eventuell mehrfach), oder alle Teilnehmenden eines Tests bzw. einer Testinstanz, die die Aufgabe bearbeitet haben. Falls alle Teilnehmenden, die die Aufgabe bearbeitet haben, gewählt wurde ergibt sich eine Einschränkung bei den Analysemöglichkeiten. Es kann keine Trennschärfe ermittelt werden, da die Rohwerte verschiedener Tests nicht vergleichbar sind. Die Analyse unterscheidet sich nach dem Aufgabentyp.

6.2.4.1 Multiple-Choice-Aufgabe

Bei einer Multiple-Choice-Aufgabe wird der Schwierigkeitsindex sowie die Trennschärfe berechnet. Zusätzlich wird eine Distraktorenanalyse durchgeführt. In Abbildung 6.4 a) ist eine solche Analyse dargestellt.

6.2.4.2 Ergänzungsaufgabe

Bei Ergänzungsaufgaben wird zu Analyse-Zwecken der Schwierigkeitsindex sowie die Trennschärfe angezeigt. Zusätzlich wird eine Liste mit allen von den Teilnehmenden eingegebenen Worten sowie die Häufigkeit der Nennungen angezeigt. Abbildung 6.4 b) zeigt die entsprechende Darstellung.

6.2.5 Testanalyse

Die Testanalyse kann für alle Instanzen eines Tests oder eine einzelne Testinstanz durchgeführt werden. Es können nur bereits vollständig bearbeitete Testinstanzen analysiert werden.

In der Testanalyse können die Rangordnung, die Prozentrangplatzierung sowie der Standardwert (siehe Abschnitt 2.6.3.1) der Teilnehmenden angezeigt werden. Bei Tests mit Aufgabengenerierung, bei dem alle Teilnehmenden die gleichen Aufgaben bearbeiten, kann zusätzlich der Reliabilitätskoeffizient berechnet werden. Diese Berechnung ist nur sinnvoll, wenn alle Teilnehmenden die gleichen Aufgaben bearbeiten. Die Oberflächen zum Anzeigen

der Rangordnung, des Prozentrangs und des Standardwerts sind in Abbildung 6.5 dargestellt. Bei der Ermittlung des Reliabilitätskoeffizienten wird lediglich ein Wert angezeigt. Dies wird nicht graphisch veranschaulicht.

Kapitel 7

Die SQL-Anfragen

In diesem Kapitel werden die SQL-Anfragen beschrieben, die die Werkzeuge auf der Datenbank ausführen. Das Kapitel sollte nicht als Einführung in SQL verstanden werden. Es werden nicht alle verwendeten Anfragen beschrieben, sondern nur die wesentlichen.

7.1 Anfragen für WETED

Für die Durchführung sind die einzelnen Attribute des Tests und der Testinstanz von Bedeutung. Es muß aus der Datenbank gelesen werden, welche Aufgabengebiete berücksichtigt und welche Schwierigkeitsstufen verwendet werden. Diese Anfragen werden nicht beschrieben. Aufgrund dieser Angaben müssen die geeigneten Aufgaben aus der Datenbank gestimmt werden, aus denen Aufgaben für die Durchführung generiert werden. Die Anfrage nach den Attributen der verwendeten Aufgaben wird nicht erläutert. Mit der Beendigung des Tests muß der Rohwert für diesen Testdurchlauf bestimmt und in die Datenbank gespeichert werden.

7.1.1 Auswahl der Testaufgaben

Für die Generierung der Testfragen ist es erforderlich, für jedes verwendete Aufgabengebiet und jeder in diesem Aufgabengebiet verwendeten Schwierigkeitsstufe die Aufgaben samt der zu vergebenden Punkte zu ermitteln. Es wird eine Anfrage benötigt, die aufgrund eines vorgegebenen Aufgabengebiets (*AG1*) sowie einer Schwierigkeitsstufe (*SS1*) alle darin enthaltenen Aufgaben sowie deren Punkte auswählt. Eine mögliche SQL-Anfrage dazu lautet:

```
SELECT WWWT_Aufgabe.AufgabenNr,Punkte FROM WWWT_Aufgabe,WWWT_Au_liegtin_AG
WHERE WWWT_Aufgabe.SSNr      =:SS1
AND   WWWT_Au_liegtin_AG.AGNr=:AG1
AND   WWWT_Aufgabe.AufgabenNr=WWWT_Au_liegtin_AG.AufgabenNr;
```

Abbildung 7.1 SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets

Diese Anfrage berücksichtigt allerdings noch nicht vorherige Ergebnisse eines Testkandidaten sowie bereits in dieser Testinstanz beantwortete Aufgaben des Kandidaten, die aufgrund eines Ausfalls des Testprogramms dem Werkzeug nicht mehr direkt zur Verfügung stehen, aber in der Datenbank gespeichert wurden. Die obige Anfrage kann dazu verwendet werden, um bei einem *Test mit Aufgabengenerierung, bei dem alle Teilnehmenden die gleichen Aufgaben gestellt bekommen*, die Aufgaben festzulegen. In diesem Fall werden keine vorherigen Ergebnisse der Teilnehmenden berücksichtigt.

Bei *CAT-Tests* und *Tests mit Aufgabengenerierung, bei dem für jede Teilnehmende eigene Aufgaben generiert werden*, sind Informationen über vorherige Ergebnisse des Teilnehmenden erforderlich.

Sowohl für Tests im Übungsmodus (besonderes häufige Verwendung vorher falsch beantworteter Aufgaben) als auch für Tests im Testmodus (möglichst keine Verwendung bereits bearbeiteter Aufgaben) wird das Wissen um eine vorherige Bearbeitung der Aufgaben durch den Teilnehmenden benötigt.

Eine Anfrage, die alle Aufgaben liefert, die von dem Teilnehmenden (*Teilnehmender1*) noch nie (innerhalb der Datenbank) bearbeitet wurden, kann folgendermaßen verwirklicht werden:

```
WITH speicher (AufgabenNr) AS
  SELECT AufgabenNr FROM WWWT_Testprotokoll
  WHERE TNr =:Teilnehmender1
SELECT WWWT_Aufgabe.AufgabenNr,Punkte FROM WWWT_Aufgabe,WWWT_Au_liegtin_AG
  WHERE WWWT_Aufgabe.SSNr      =:SS1
  AND   WWWT_Au_liegtin_AG.AGNr=:AG1
  AND   WWWT_Aufgabe.AufgabenNr=WWWT_Au_liegtin_AG.AufgabenNr
  AND   WWWT_Aufgabe.AufgabenNr NOT IN speicher;
```

Abbildung 7.2 SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets unter Ausschluß vorher bearbeiteter Aufgaben

Die Anfrage nach Aufgaben, die bereits bearbeitet wurden, allerdings nicht in der selben Testinstanz (*TI1* und *Test1*), lautet:

```
WITH speicher (AufgabenNr) AS
  SELECT AufgabenNr FROM WWWT_Testprotokoll
  WHERE TNr =:Teilnehmender1
  AND   (TINr=!:TI1 OR TestNr=!:Test1)
SELECT WWWT_Aufgabe.AufgabenNr,Punkte FROM WWWT_Aufgabe,WWWT_Au_liegtin_AG
  WHERE WWWT_Aufgabe.SSNr      =:SS1
  AND   WWWT_Au_liegtin_AG.AGNr=:AG1
  AND   WWWT_Aufgabe.AufgabenNr=WWWT_Au_liegtin_AG.AufgabenNr
  AND   WWWT_Aufgabe.AufgabenNr IN speicher;
```

Abbildung 7.3 SQL-Anfrage zur Bestimmung aller Aufgaben einer Schwierigkeitsstufe innerhalb eines Aufgabengebiets, die bereits von Teilnehmenden bearbeitet wurden

Beim Testmodus ist nicht von Bedeutung, wie und wie oft die bereits bearbeiteten Aufgaben vom Teilnehmenden gelöst wurden. Im Übungsmodus muß zusätzlich für jede bereits bearbeitete Aufgabe für jeden Lösungsversuch des Teilnehmenden die Zeit sowie die Qualität der Antwort (erreichte Punkte/max. erreichbare Punkte) bestimmt werden. Entsprechend dieser Werte erhöht oder verringert sich die Wahrscheinlichkeit, die Frage zu verwenden. Eine Anfrage für eine Aufgabe *Aufgabe1* kann so aussehen:

```
SELECT (max_Punktzahl/erreichte_Punkt_Z) AS qual, Zeitpunkt
FROM WWWT_Testprotokoll
WHERE AufgabenNr=:Aufgabe1
AND   TNr           =:Teilnehmender1
ORDER BY Zeitpunkt;
```

Abbildung 7.4 SQL-Anfrage zur Bestimmung des Zeitpunktes aller Antworten eines Teilnehmenden zu einer Aufgabe sowie der Qualität der Antwort

Aufgrund obiger Anfragen kann für alle unterstützten Test-Typen eine Auswahl der zu generierenden Aufgaben getroffen werden.

7.1.2 Ermittlung des Rohwerts

Nachdem die Korrektur des Rohwerts bereits in der Relation *WWWT_Testprotokoll* bei der Bearbeitung der einzelnen Aufgaben vorgenommen wird (siehe 6.1.2), ist für die Ermittlung des Rohwerts eines Teilnehmenden lediglich die Aufsummierung aller erhaltenen Punkte (es können u.U. auch negative Punkte vergeben werden) der einzelnen Aufgaben erforderlich.

Die Anfrage dazu lautet:

```
SELECT SUM(erreichte_Punkt_Z) FROM WWWT_Testprotokoll
WHERE   TNr           =:Teilnehmender1
AND     TINR          =:Testinstanz1
AND     TestNr        =:Test1
```

Abbildung 7.5 SQL-Anfrage, die den Rohwert von *Teilnehmender1* in der Testinstanz *Testinstanz1* mit der Testnummer *Test1* bestimmt

7.2 Anfragen für WEBEWAU

7.2.1 Bewertung der Ergebnisse

Vor der Bewertung der Ergebnisse müssen zuerst alle Testinstanzen gesucht werden, in der eine Bewertung vorgenommen werden kann. Lediglich Testinstanzen im Testmodus (**Modus** muß den Wert **T** besitzen) dürfen bewertet werden. Eine Bewertung kann durchgeführt werden, wenn noch keine Bewertung in der Testinstanz vorgenommen wurde und entweder der Bearbeitungszeitraum bereits überschritten wurde oder alle Teilnehmenden den Test

bearbeitet und einen Rohwert erhalten haben. Bei der Anfrage wird davon ausgegangen, daß alle Teilnehmenden einer Testinstanz gleichzeitig bewertet werden. Die gleichzeitige Bewertung aller Teilnehmenden ergibt sich aus der Anforderung nach einer objektiven Bewertung. Für alle Teilnehmenden einer Testinstanz müssen die gleichen Bewertungskriterien verwendet werden.

```
SELECT DISTINCT TestNr, TINr FROM WWWT_Testinstanz AS TI
WHERE Modus = 'T'
AND NOT EXISTS (SELECT * FROM WWWT_Teilnahme AS TN
                WHERE TN.TestNr= TI.TestNr
                AND TN.TINr = TI.TINr
                AND TN.Note IS NOT NULL)
AND ( (TI.End_Datum<:aktuellesDatum
      OR (TI.End_Datum=:aktuellesDatum AND End_Zeit<:aktuelleZeit))
      OR NOT EXISTS (SELECT * FROM WWWT_Teilnahme AS TN
                    WHERE TN.TestNr= TI.TestNr
                    AND TN.TINr = TI.TINr
                    AND TN.Rohwert IS NULL))
GROUP BY TestNr
```

Abbildung 7.6 SQL-Anfrage zur Bestimmung aller Testinstanzen, bei denen eine Bewertung vorgenommen werden kann

Die Auswahl aller Rohwerte einer Testinstanz bzw. aller Testinstanzen eines Tests (bei einer normorientierten Bewertung unter Berücksichtigung aller Testinstanzen), sowie aller Notenschlüssel, die ein bestimmtes Kriterium erfüllen, wird nicht explizit aufgeführt.

7.2.2 Aufgabenanalyse

Die Auswahl der Aufgaben kann nach verschiedenen Kriterien erfolgen. Exemplarisch wird eine Anfrage gezeigt, die aufgrund einer Testinstanz und einem Suchstring für den Fragetext die entsprechenden Aufgaben zurückgibt.

```
SELECT AufgabenNr, Fragestellung FROM WWWT_Aufgabe
WHERE Fragestellung LIKE '%Suchstring%'
AND AufgabenNr IN (SELECT AufgabenNr FROM WWWT_Testprotokoll
                  WHERE TN.TestNr= :TestNr
                  AND TN.TINr = :TINr
```

Abbildung 7.7 SQL-Anfrage zur Bestimmung aller Aufgaben für die Aufgabenanalyse aufgrund vorgegebenem Suchstring und Testinstanz (TestNr und TINr)

7.2.2.1 Schwierigkeitsindex

Der Schwierigkeitsindex kann für alle Aufgabentypen gleich berechnet werden. Bei der Berechnung kann eine Einschränkung bezüglich der berücksichtigten Teilnehmenden erfolgen.

Es können entweder alle Teilnehmenden berücksichtigt werden (bei mehrfacher Bearbeitung mehrmals) oder eine Beschränkung auf Teilnehmende eines Tests bzw. einer Testinstanz erfolgen.

Für die Berechnung des Schwierigkeitsindizes wird folgende Formel verwendet:

$$\text{Schwierigkeitsindex} = \frac{\sum_{i=1}^N (\text{erreichte Punkte}/\text{max. erreichbare Punkte})}{N} * 100$$

, wobei N die Anzahl der Bearbeitungen ist. Bei dieser Formel ist darauf zu achten, daß bei der erreichten Punktzahl als kleinster Wert Null verwendet wird, also negativen Werte in Null umgewandelt werden.

Dieses Verfahren kann sowohl für Aufgaben mit einer zweiwertigen als auch mit einer mehrwertigen Punktvergabe verwendet werden. Ebenso hat eine unterschiedliche Gewichtung bei verschiedenen Tests (z.B. eine höhere Punktvergabe bei einem Test) keinen Einfluß auf das Ergebnis des Schwierigkeitsindizes. Die folgende Anfrage gibt den Schwierigkeitsindex einer Aufgabe für eine bestimmte Testinstanz zurück.

```
SELECT 100*(SUM((CASE WHEN erreichte_punkt_z>0 then erreichte_punkt_z
                ELSE 0
                END)/max_Punktzahl)/COUNT(max_Punktzahl)) AS Schw_Index
FROM WWWT_Testprotokoll
WHERE AufgabenNr =:AufgabenNr
AND   TestNr     =:TestNr
AND   TINr       =:TINr
```

Abbildung 7.8 Berechnung des Schwierigkeitsindizes für eine Aufgabe (**AufgabenNr**) bezogen auf eine Testinstanz (**TestNr** und **TINr**)

Falls ein Test die Bezugsgruppe bildet, wird die letzte Bedingung weggelassen (**TINr = :TINr**), für alle Teilnehmenden, die die Aufgabe bearbeitet haben, auch die vorletzte (**TestNr =:TestNr**).

7.2.2.2 Trennschärfe

Zur Berechnung der Trennschärfe wird zunächst die Standardabweichung des Rohwerts, der durchschnittliche Rohwert sowie die Anzahl der Teilnehmenden benötigt.

```
SELECT (SQRT((COUNT(Rohwert)*SUM(Rohwert*Rohwert))
            -(SUM(Rohwert)*SUM(Rohwert)))/COUNT(Rohwert)) AS X_s ,
       (SUM(Rohwert)/COUNT(Rohwert)) AS X_quer , COUNT(Rohwert) AS N
FROM WWWT_Teilnahme
WHERE TestNr     =:TestNr
AND   TINr       =:TINr
```

Abbildung 7.9 Berechnung der Standardabweichung des Rohwerts, des durchschnittlichen Rohwerts und der Anzahl der Teilnehmenden

Die Standardabweichung wird als X_s bezeichnet, der durchschnittliche Rohwert als X_{quer} und die Anzahl als N ¹.

Zusätzlich wird die Standardabweichung bei der Punktvergabe der Aufgabe und die durchschnittliche Punktzahl der Aufgabe benötigt.

```
SELECT (SQRT((COUNT(erreichte_Punkt_Z)*
              SUM(CAST(erreichte_Punkt_Z/max_Punktzahl AS DOUBLE)*
                  CAST(erreichte_Punkt_Z/max_Punktzahl AS DOUBLE)))
          -(SUM(CAST(erreichte_Punkt_Z/max_Punktzahl AS DOUBLE))*
            SUM(CAST(erreichte_Punkt_Z/max_Punktzahl AS DOUBLE))))
        /COUNT(erreichte_Punkt_Z)) AS P_s,
(SUM(erreichte_Punkt_Z/max_Punktzahl)/COUNT(erreichte_Punkt_Z)) AS P_quer
FROM WWWT_Testprotokoll
WHERE TestNr      =:TestNr
AND   TINr        =:TINr
AND   AufgabenNr =:AufgabenNr
```

Abbildung 7.10 Berechnung der Standardabweichung der Punktzahl und der durchschnittlichen Punktzahl einer Aufgabe

Die Standardabweichung der Punktzahl wird als P_s und die durchschnittliche Punktzahl als P_{quer} bezeichnet.

Die Konvertierung des Quotienten `erreichte_Punkt_Z/max_Punktzahl` in einen Wert vom Typ `DOUBLE` ist erforderlich, da das verwendete Datenbanksystem die Berechnung sonst mit einer Fehlermeldung abbricht.

Mit einer weiteren SQL-Anfrage unter Verwendung der obigen Werte kann die Trennschärfe einer Aufgabe bestimmt werden.

```
SELECT (SUM(((CAST(erreichte_Punkt_Z/max_Punktzahl AS DOUBLE)-
                  :P_quer)/:P_s)*((Rohwert-:X_quer)/:X_s)))/:N) AS Trennschaerfe
FROM WWWT_Teilnahme AS TN, WWWT_Testprotokoll AS TP
WHERE TN.TestNr      =:TestNr
AND   TN.TINr        =:TINr
AND   TN.TestNr      =TP.TestNr
AND   TN.TINr        =TP.TINr
AND   TN.TNr         =TP.TNr
AND   TP.AufgabenNr =:AufgabenNr
```

Abbildung 7.11 Bestimmung der Trennschärfe

Bei der Anfrage muß beachtet werden, daß keine Division durch Null erfolgt². Falls beispielsweise alle Teilnehmenden die Aufgabe gleich beantworten, ergibt P_s den Wert Null.

¹Diese Angaben werden für spätere SQL-Anfragen benötigt.

²Dies kann entweder bei der Integration der SQL-Anfragen zu einer Anfrage oder beim Anwendungsprogramm erfolgen.

Der Trennschärfekoeffizient beträgt in diesem Fall 0,00.

Falls die Trennschärfe für alle Testinstanzen eines Tests berechnet werden soll, entfällt bei den obigen Anfragen jeweils die Einschränkung auf eine Testinstanz.

Die Verwendung des Quotienten (`erreichte_Punkt_Z/max_Punktzahl`) anstatt des Wertes `erreichte_Punkt_Z` garantiert ein korrektes Ergebnis, auch wenn bei der Aufgabengenerierung eine Reduzierung der maximalen Punktzahl für die Aufgabe durchgeführt wurde.

7.2.2.3 Distraktorenanalyse

Bei der Distraktorenanalyse wird für jedes Item der zu analysierenden Aufgabe gezählt, wie oft es für eine bestimmte Testinstanz in `WWWT_gew_Loesung` ausgewählt wurde.

```
SELECT I.Position, COUNT(I.Position) AS Nennungen
FROM WWWT_Item AS I, WWWT_Gew_Loesung AS GL
WHERE GL.TestNr      =:TestNr
AND   GL.TINr       =:TINr
AND   GL.AufgabenNr =:AufgabenNr
AND   I.AufgabenNr  =:AufgabenNr
AND   I.Position    =CAST(gew_Loesung AS int)
GROUP BY I.Position
```

Abbildung 7.12 Anzahl der Nennungen der Items einer Aufgabe

Bei der Anfrage werden keine Werte für nie gewählte Items ermittelt. Für diese muß als Anzahl der Nennungen Null gewählt werden.

Zusätzlich wird die Trennschärfe für jedes Item ermittelt. Zur Ermittlung des durchschnittlichen Rohwerts, der Standardabweichung des Rohwerts sowie der Anzahl der Teilnehmenden kann die gleiche Anfrage wie für die Berechnung der Trennschärfe der Aufgabe verwendet werden, bzw. die berechneten Werte. Für die weitere Berechnung wird als Punktwert eins gewählt, falls das Item ausgewählt wurde, ansonsten Null. Für die Berechnung der Standardabweichung und der durchschnittlichen Punktzahl kann der bereits ermittelte Wert über die Anzahl der Nennungen dieses Items verwendet werden. Im folgenden wird dieser Wert *Itemsomme* genannt. Die durchschnittliche Punktzahl wird mit $\frac{\text{Itemsomme}}{N}$ berechnet, die Standardabweichung mit:

$$P_s = \frac{\sqrt{N * \text{Itemsomme} - \text{Itemsomme}^2}}{N} \quad (7.1)$$

Der Trennschärfekoeffizient wird mittels zwei weiteren Anfragen berechnet. Die erste Anfrage berechnet den Teil der Summe aus Formel 2.14, bei dem die Teilnehmenden das Item gewählt haben, die zweite Anfrage berücksichtigt die Teilnehmenden, die das Item nicht gewählt haben. Die erste Anfragen lautet:

```

SELECT (SUM((1-:P_quer)/:P_s)*((Rohwert-:X_quer)/:X_s)) AS Summe1
FROM WWWT_Gew_Loesung AS GL, WWWT_Teilnahme AS TN
WHERE GL.TestNr=:TestNr
AND   GL.TINr   =:TINr
AND   GL.AufgabenNr=:AufgabenNr
AND   GL.TNr    =TN.TNr
AND   GL.TINr   =TN.TINr
AND   GL.TestNr=TN.TestNr
AND   CAST(GL.Gew_Loesung AS Int)=:PositionDesItems

```

Abbildung 7.13 Bestimmung eines Teils der Formel zur Berechnung der Trennschärfe (Item gewählt)

Es muß bei der Anfrage darauf geachtet werden, daß keine Division durch Null erfolgt. Falls *P_s* oder *X_s* Null sind, kann der Trennschärfekoeffizient nicht ermittelt werden, bzw ergibt einen Wert von 0,00.

```

SELECT (SUM(((1-:P_quer)/:P_s)*((Rohwert-:X_quer)/:X_s))) AS Summe2
FROM WWWT_Teilnahme AS TN
WHERE TN.TestNr=:TestNr
AND   TN.TINr   =:TINr
AND   NOT EXISTS (SELECT * FROM WWWT_Gew_Loesung AS GL
                  WHERE GL.TestNr      =TN.TestNr
                  AND   GL.TINr        =TN.TINr
                  AND   GL.TNr         =TN.TNr
                  AND   GL.AufgabenNr =:AufgabenNr
                  AND   CAST(GL.gew_Loesung AS Int)=:PositionDesItems)

```

Abbildung 7.14 Bestimmung eines Teils der Formel zur Berechnung der Trennschärfe (Item nicht gewählt)

Die ermittelten Werte (*Summe1* und *Summe2*) werden addiert und durch *N* dividiert und liefern so den Trennschärfekoeffizienten.

7.2.2.4 Gewählte Lösungsworte

Die verschiedenen gewählten Lösungsworte werden mit einer Anfrage auf der Relation *WWWT_gew_Loesung* (mit *GROUP BY gew_Loesung*) ermittelt, wobei ebenfalls die Anzahl gezählt wird.

7.2.3 Testanalyse

Zur Bestimmung der Rangordnung wird eine einfache, nach dem Rohwert sortierte Anfrage auf *WWWT_Teilnahme* ausgeführt. Die Berechnung der Prozentrangplatzierung kann

ebenfalls aufgrund dieser Daten erfolgen (siehe Abschnitt 2.6.3.1). Zur Bestimmung der Standardwerte wird zusätzlich die Standardabweichung benötigt. Die Ermittlung der Standardabweichung wird bereits bei der Trennschärfe beschrieben.

7.2.3.1 Reliabilitätskoeffizient

Es gibt verschiedene Methoden zur Bestimmung des Reliabilitätskoeffizienten (siehe Abschnitt 2.2.2). Einige Methoden verlangen die Durchführung eines weiteren Tests durch dieselben Teilnehmenden. Dies kann derselbe Test (*Test-Retest-Reliabilität*) oder ein anderer Test (*Paralleltest-Reliabilität*) sein. Die Durchführung eines weiteren Tests ist in der Praxis jedoch oft nicht möglich. Deshalb wird zur Bestimmung des Reliabilitätskoeffizienten ein Verfahren zur Messung der *inneren Konsistenz* gewählt, bei dem ein Testdurchlauf ausreicht. Das Verfahren der *Konsistenzanalyse* setzt voraus, daß alle Aufgaben das gleiche Persönlichkeitsmerkmal messen. Davon kann in einem Test, der in mehrere Aufgabengebiete unterteilt ist, jedoch nicht ausgegangen werden. Als Verfahren zur Bestimmung des Reliabilitätskoeffizienten wird die *Testhalbierung* verwendet. Bei diesem Verfahren wird der Test in zwei gleichwertige Hälften geteilt. Durch die Einteilung der Aufgaben in Schwierigkeitsstufen und Aufgabengebiete kann die Halbierung automatisch nach diesen Kriterien erfolgen.

Die *Aufteilung in zwei gleichwertige Testhälften* erfolgt jeweils für eine Schwierigkeitsstufe innerhalb eines Aufgabengebiets. Die Anfrage zur Bestimmung der Aufgaben lautet:

```
SELECT A.AufgabenNr, A.Punkte, A.SSNr
FROM WWWT_Aufgabe AS A, WWWT_Aufg_vorgabe AS AV
WHERE AV.TestNr      =:TestNr
AND   AV.TINr       =:TINr
AND   A.AufgabenNr =AV.AufgabenNr
AND   A.AufgabenNr IN (SELECT AufgabenNr FROM WWWT_Au_liegtin_AG
                        WHERE AGNr=:AGNr)

GROUP BY SSNr
```

Abbildung 7.15 Auswahl der Aufgaben zur Testhalbierung

Die Aufgaben werden entsprechend ihrer Punktzahl in zwei Hälften geteilt. Da eine genaue Einteilung in zwei Hälften nicht immer möglich ist, wird zusätzlich der relative Anteil der Aufgaben der beiden Teile am Gesamttest bestimmt. Anschließend wird die Korrelation der zwei Testhälften nach folgender Formel bestimmt (aus [LiRa94]):

$$r_{12} = \frac{N * \sum_{i=1}^N (X_{1i} * X_{2i}) - \sum_{i=1}^N X_{1i} * \sum_{i=1}^N X_{2i}}{\sqrt{\left(N \sum_{i=1}^N X_{1i}^2 - \left(\sum_{i=1}^N X_{1i}\right)^2\right) \left(N \sum_{i=1}^N X_{2i}^2 - \left(\sum_{i=1}^N X_{2i}\right)^2\right)}} \quad (7.2)$$

Diese Formel, angewandt auf die Datenbank, ergibt die Anfrage:

```

WITH ErgebnisTest1 AS
(SELECT  SUM(erreichte_Punkt_Z) AS X1,TNr FROM WWWT_Testprotokoll AS TP
  WHERE TP.TestNr=:TestNr AND TP.TINr=:TINr
  AND   TP.AufgabenNr IN (:Nr1,:Nr2,:Nr3)--alle AufgabenNr des ersten Teils
  GROUP BY TNr),
ErgebnisTest2 AS
(SELECT  SUM(erreichte_Punkt_Z) AS X2,TNr FROM WWWT_Testprotokoll AS TP
  WHERE TP.TestNr=:TestNr AND TP.TINr=:TINr
  AND   TP.AufgabenNr IN (:Nr4,:Nr5,:Nr6)--alle AufgabenNr des zweiten Teils
  GROUP BY TNr)
SELECT ((COUNT(T1.TNr)*SUM(X1*X2)-SUM(X1)*SUM(X2))/
  (SQRT((COUNT(T1.TNr)*SUM(X1*X1)-SUM(X1)*SUM(X1))*
    (COUNT(T1.TNr)*SUM(X2*X2)-SUM(X2)*SUM(X2)))))) AS r_12
FROM ErgebnisTest1 AS T1, ErgebnisTest2 AS T2
WHERE T1.TNr = T2.TNr

```

Abbildung 7.16 Bestimmung der Korrelation zweier Testhälften

Aufgrund dieser Korrelation wird nach der Formel von HORST (entnommen aus [LiRa94]) der Reliabilitätskoeffizient bestimmt. Die Formel lautet:

$$r_{tt} = \frac{r_{12} \left(\sqrt{r_{12}^2 + 4pq(1 - r_{12}^2)} - r_{12} \right)}{2pq(1 - r_{12}^2)} \quad (7.3)$$

wobei p und q die relativen Anteile der beiden Teile am Test sind.

Kapitel 8

Erfahrungen bei der Durchführung von rechnerunterstützten Tests

Um Erfahrungen im Bereich des rechnerunterstützten Testens zu sammeln, wurden in einer parallel zur Erstellung dieser Arbeit laufenden Veranstaltung bereits rechnerunterstützte Tests durchgeführt. Bei dieser Veranstaltung handelte es sich um das Hauptseminar *Multimediale Anwendungen im Internet*.

8.1 Informationen über die Veranstaltung

Die Lehrveranstaltung *Multimediale Anwendungen im Internet* wurde im Wintersemester 1997/98 zusammen von der Universität Stuttgart und der Universität Tübingen durchgeführt (siehe [HSMMA98]). In Stuttgart beteiligte sich die Abteilung Verteilte Systeme des Instituts für Parallele und Verteilte Höchstleistungsrechner (IPVR) der Fakultät Informatik im Rahmen eines Hauptseminars an dieser Veranstaltung, in Tübingen der Arbeitsbereich Technische Informatik des Wilhelm-Schickard-Instituts für Informatik im Rahmen eines Proseminars. Prüfer waren Prof. Dr. Kurt Rothermel in Stuttgart und Prof. Dr. Wolfgang Rosenstiel in Tübingen. Die Betreuung erfolgte in Stuttgart durch Dipl. Inform. Frank Schiele und in Tübingen durch Dipl. Math. Julia Wunner.

Es wurden von den Teilnehmenden insgesamt neun Seminar-Vorträge durchgeführt, wobei lediglich bei sechs Vorträgen ein Test angeboten wurde. Zusätzlich wurden weitere Vorträge gehalten, um den Umgang mit den Mbone-Tools¹ zu üben.

¹Die verwendeten Werkzeuge zur Durchführung des Seminars über das Internet (siehe <http://www.mbone.de>).

8.2 Ablauf der Testdurchführung

Die Teilnehmenden wurden gebeten, vor ihrem Vortrag 3-5 Fragen für einen Eingangs- und einen Abschlußtest zur Verfügung zu stellen². Bei den Fragen konnte es sich um Multiple-Choice oder Ergänzungsaufgaben handeln. Einige Teilnehmende verwendeten für Eingangs- und Abschlußtests die gleichen Fragen, bei anderen unterschieden sich die Fragen. Der Eingangstest eines Vortrags wurde von den Zuhörenden unmittelbar vor dem Vortrag ausgefüllt, der Abschlußtest nach dem Vortrag und anschließender Diskussion. Die Ergebnisse des Tests konnten jederzeit über ein Java-Applet eingesehen werden.

8.3 Technische Durchführung

Die Durchführung des rechnerunterstützten Testens im Rahmen des Hauptseminars galt in erster Linie dem Erfahrungsgewinn in diesem Bereich. Insbesondere sollte keine Bewertung der Teilnehmenden erfolgen. Um dies zu verhindern, wurden die Daten ohne einen Bezug zum Teilnehmenden gespeichert. Damit die Information, wer welchen Test ausgeführt hat, für die Auswertung nicht gänzlich verloren ging, wurde bei der ersten Bearbeitung eines Teilnehmenden ein Cookie³ gesetzt, das bei weiteren Bearbeitungen als Benutzererkennung verwendet werde. Als Cookie-Inhalt wurde lediglich ein Zeitstempel verwendet, so daß kein Rückschluß auf den Teilnehmenden möglich war.

Aufgrund obiger Voraussetzungen konnte zum einen auf eine Aufsicht verzichtet werden und zum anderen konnten die Fragen frei zugänglich gemacht werden.

Da die Durchführung parallel zur Erstellung dieser Arbeit erfolgte, waren zu Beginn weder die geeigneten Werkzeuge noch ein Datenbankschema entworfen. Es wurde ein minimales Schema zum Sammeln und Speichern der Daten gewählt. Die Fragen wurden nicht in einem Java-Applet sondern als HTML-Formular zur Verfügung gestellt. Ein serverseitiges Java-Programm (Servlet) trug diese Daten in eine Datenbank ein. Die Ergebnisse der Tests konnten mit einem Java-Applet eingesehen werden.

Es wurde ein Datenbank-Schema mit drei Relationen verwendet. Zwei Relationen enthielten Informationen über die Aufgaben bzw. die richtigen Lösungen, die von dem Applet zum Anzeigen der Ergebnisse verwendet wurden. In einer weiteren Relation wurden die gewählten Lösungen der Teilnehmenden eingetragen.

²An dieser Stelle möchte ich mich bei den Teilnehmenden für die rege Teilnahme, sowohl bei der Bereitstellung der Fragen als auch bei der Bearbeitung der Tests bedanken

³Genauere Informationen zum Thema Cookie unter:
http://search.netscape.com/newsref/std/cookie_spec.html

8.4 Ergebnisse

8.4.1 Tests

Es wurden bei sechs Vorträgen Eingangs- und Abschlußtests durchgeführt, also insgesamt 12 Tests. 9 Tests wurden von einigen Teilnehmenden doppelt bearbeitet, wodurch sich eine Anzahl von 21 Testinstanzen ergab. Die doppelte Bearbeitung ist darauf zurückzuführen, daß einige Teilnehmenden das Hauptseminar an dem gleichen Arbeitsplatzrechner verfolgten und die Tests entsprechend nacheinander ausführten. Dabei wurde jeweils das gleiche Cookie verwendet.

8.4.2 Teilnehmende

Laut Datenbank haben 47 verschiedene Teilnehmende einen Test bearbeitet. Die einzelnen Tests wurden allerdings nur zwischen 5 und 16 mal bearbeitet, im Durchschnitt somit 9,3 mal. Dies liegt etwas unter dem Durchschnitt der Zuhörenden der Vorträge. Es ist zu beobachten, daß viele Teilnehmenden-Nummern nur bei einem oder zwei Tests auftauchen. Lediglich drei Teilnehmenden-Nummern sind bei (fast) allen Tests zu finden. Offensichtlich ist der Versuch gescheitert, eine Zuordnung der Teilnehmenden mit einem Cookie vorzunehmen. Die Ursachen hierfür liegen darin, daß Teilnehmende in ihrem WWW-Browser keine Cookies zulassen (dies ist als Option einstellbar) oder den Browser von verschiedenen Accounts aus gestartet haben. Aufgrund der fehlenden Zuordnung der Teilnehmenden über einen Test hinaus werden keine Analysen in diesem Zusammenhang vorgenommen.

8.4.3 Aufgaben

Durch die Tests sind insgesamt 37 verschiedene Aufgaben angeboten, wobei einige Aufgaben sowohl im Eingangs-, als auch im Abschlußtest genannt wurden. Unter den Aufgaben befanden sich 23 Multiple-Choice-Aufgaben mit Mehrfachnennung, 12 ohne Mehrfachnennung sowie zwei Ergänzungsaufgaben.

Bei den Multiple-Choice-Aufgaben ohne Mehrfachnennung wurden die Aufgaben allerdings 9 mal in einer vom Entwurf nicht vorgesehenen Weise verwendet. In diesen Aufgaben wurde kein Wissen abgefragt, sondern die persönlichen Erfahrungen der Teilnehmenden bestimmt. Im folgenden werden sie *Umfrage-Aufgaben* genannt. Als Beispiel hierfür folgende Frage:

Zum Thema "Teleteching" habe ich vor diesem Seminar :	
<input type="radio"/> noch nie etwas gehört.	(keine Nennung)
<input type="radio"/> schon mal gehört, aber nichts genaues.	(7 Nennungen)
<input type="radio"/> weiss ich bestens Bescheid.	(4 Nennungen)

Abbildung 8.1: Umfrage-Aufgabe

Diese Aufgaben lassen sich nur schwer in den Datenbankentwurf dieser Arbeit integrieren. In der beim Hauptseminar verwendeten Datenbank wurden alle Items dieser Art von Aufgaben als richtig gewertet. Im Datenbankentwurf dieser Arbeit ist das nicht möglich, da für eine Multiple-Choice-Aufgabe ohne Mehrfachnennung nur ein richtiges Item erlaubt ist. Statt dessen werden alle Items als falsch bewertet und für Umfrage-Aufgaben Null Punkte vergeben⁴. Diese Aufgaben werden damit bei keiner automatischen Testgenerierung verwendet. Lediglich bei Tests, bei denen alle Teilnehmenden die gleichen Aufgaben bearbeiten, kann eine solche Aufgabe verwendet werden, falls sie in die entsprechende Relation `WWWT_Aufg_vorgabe` eingetragen wird. Bei den folgenden Analysen werden diese Aufgaben nicht weiter betrachtet. Für die Durchführung während des Hauptseminars waren die Umfrage-Aufgaben durchaus interessant, um dem Vortragenden Erkenntnisse über den Wissensstand zu geben. Zur Bewertung der Teilnehmenden bzw. der Analyse der Aufgaben und Tests können sie jedoch nicht verwendet werden.

8.4.3.1 Analyse der Aufgaben

Der Schwierigkeitsindex der Aufgaben reichte von 14% bis 100%. Die Berechnung des Indizes bezog sich jeweils auf einen Test, d.h. für Aufgaben, die in zwei Tests vorkamen, wurden zwei Indizes bestimmt. Die folgende Tabelle zeigt die Verteilung der Schwierigkeitsindizes:

Anzahl der Aufgaben	Wertebereich des Schwierigkeitsindizes
3	< 30%
3	30% - 50%
10	50% - 70%
13	70% - 90%
10	> 90%

Der Durchschnittliche Schwierigkeitsindex liegt damit bei 71%.

Werden Eingangs- und Abschlußtests getrennt betrachtet, ergibt sich für Eingangstests ein durchschnittlicher Schwierigkeitsindex von 67% sowie 75% bei Abschlußtests.

Anhand dieser Ergebnisse zeigt sich, daß der Schwierigkeitsindex der Aufgaben weit gestreut ist und sich gut für die Einteilung in verschiedene Schwierigkeitsstufen eignet.

Bei den Aufgabentypen läßt sich feststellen, daß die Ergänzungsaufgaben mit einem durchschnittlichen Schwierigkeitsindex von 22% nur schwer zu lösen waren, wohingegen Multiple-Choice-Aufgaben mit Mehrfachnennung (70%) und ohne Mehrfachnennung (77%) in etwa die gleiche Schwierigkeit besaßen. Allerdings ist diese Aussage durch die Verwendung von insgesamt nur zwei Ergänzungsaufgaben mit Vorsicht zu betrachten.

⁴Dies erforderte eine kleine Anpassung einer Check-Constraints, bei der vorher als Bedingung `Punkte > 0` eingetragen war.

Die Trennschärfe wird jeweils für eine Aufgabe eines Tests berechnet, wobei dies nur bei Aufgaben, möglich ist, bei denen verschiedene Lösungen gewählt wurden. Wurden immer die gleichen Lösungen gewählt (beispielsweise bei einem Schwierigkeitsindex von 100%), kann nicht zwischen leistungsstarken und -schwachen Teilnehmenden unterschieden werden und der Trennschärfekoeffizient beträgt 0,00 (Dies war bei 17 Aufgaben eines Tests der Fall, darunter auch die Umfrage-Aufgaben).

Bei den 31 berechneten Trennschärfekoeffizienten ergaben sich Werte zwischen -0,30 und 1,00. Allerdings wurde nur ein negativer Wert beobachtet. In der folgenden Tabelle ist die Verteilung der Koeffizienten dargestellt:

Anzahl der Aufgaben	Wertebereich des Trennschärfekoeffizienten
1	< 0,00
6	0,00 - 0,40
6	0,40 - 0,60
9	0,60 - 0,80
9	> 0,80

Nach [Sche80] sollten Aufgaben einen Trennschärfekoeffizienten von größer oder gleich 0,40 haben, um in Tests verwendet zu werden. Damit könnten 20 Aufgaben (vier wurden in je zwei Tests gestellt) in weiteren Tests verwendet werden. Allerdings ist hier zu bedenken, daß der Trennschärfekoeffizient in sehr kleinen Tests (3-5 Aufgaben) ermittelt wurde.

Die Distraktorenanalyse wird nicht für die Umfrage-Aufgaben durchgeführt, da sonst die Ergebnisse verfälscht werden.

Von 104 Items wurden 29 von keinem Teilnehmenden gewählt. 19 dieser Items entstammen Multiple-Choice-Aufgaben ohne Mehrfachnennung die restlichen 10 von Multiple-Choice-Aufgaben mit Mehrfachnennung. Diese Items sollten also bei weiteren Testdurchführung durch andere Items ersetzt werden. Allerdings ist auch diese Aussage dadurch eingeschränkt, daß die Aufgaben nur von wenigen Teilnehmenden bearbeitet wurden. Die entsprechenden Multiple-Choice-Aufgaben ohne Mehrfachnennung wurden bei durchschnittlich 3,9 Items lediglich 12,9 mal bearbeitet, Aufgaben mit Mehrfachnennung bei durchschnittlich 6,8 Items wurden 13,5 mal bearbeitet.

8.4.4 Genaue Analyse einer Aufgabe

Im folgenden wird die genaue Analyse einer einzelnen Aufgabe vorgenommen (siehe Abbildung 8.2). Diese Aufgabe wurde sowohl in einem Eingangstest als auch im entsprechenden Abschlußtest gestellt.

Der Schwierigkeitsindex beträgt beim Eingangstest 25%, beim Ausgangstest 40%. Die Aufgabe konnte also im Verhältnis zur Anzahl der Teilnehmenden nach dem Vortrag öfter richtig gelöst werden. Bei einer Multiple-Choice-Aufgabe ohne Mehrfachnennung mit 5 Items ergibt die zufällige Wahl eines Items noch einen Schwierigkeitsindex von 20%, so daß davon

Zu welchen der folgenden Akronyme kann unser (Haupt/Pro-)Seminar zugeordnet werden?			
<input type="radio"/>	RLP	Eingangstest: 0 mal	Abschlusstest: 3 mal
<input type="radio"/>	RIS	2 mal	2 mal
<input type="radio"/>	RAS	1 mal	0 mal
<input type="radio"/>	IHL	2 mal	0 mal
<input type="radio"/>	LRB	3 mal	0 mal

Die richtige Lösung ist RIS

Abbildung 8.2: Aufgabe für die Analyse

ausgegangen werden kann, daß vor dem Vortrag kein Teilnehmender durch sein Wissen die Aufgabe richtig beantwortet hat. Der Schwierigkeitsindex von 40% beim Abschlußtest deutet darauf hin, daß einige Teilnehmende aufgrund des Vortrags das Wissen erlangt haben, die Aufgabe richtig beantworten zu können.

Der Trennschärfekoeffizient beträgt beim Eingangstest 0,67; beim Abschlußtest 0,91. Daran läßt sich noch einmal der Zusammenhang zwischen Trennschärfe und Schwierigkeitsindex verdeutlichen. Bei einem Schwierigkeitsindex von 50% kann die höchste Trennschärfe erreicht werden.

Zusammenfassend kann gesagt werden, daß die Aufgabe sehr gut für den Abschlußtest geeignet ist (mit einer Trennschärfe von 0,91), für den Eingangstest allerdings zu schwierig war.

Kapitel 9

Zusammenfassung und Ausblick

Zusammenfassung

Innerhalb dieser Arbeit wurden die Grundlagen der pädagogischen Psychologie zur Durchführung und Bewertung von Tests erarbeitet sowie die Besonderheiten von rechnerunterstützten Tests untersucht.

Dabei hat sich gezeigt, daß die Auswahl der Aufgaben für eine rechnerunterstützte Auswertung eingeschränkt ist. Dies ist allerdings nicht als Nachteil zu werten, da so die Auswertungsobjektivität (ein wichtiges Kriterium von Tests) erreicht wird. Durch die objektive Punktvergabe für die Aufgaben und den daraus resultierenden Rohwert, der als Grundlage der Bewertung verwendet wird, wird die Auswertungsobjektivität ebenfalls erreicht. Die Durchführungsobjektivität kann nicht garantiert werden, da der Ausfall von Hard- und Software nicht ausgeschlossen werden kann. Ein Ausfall des Programms wird jedoch in der Datenbank gespeichert, so daß die Ergebnisse ggf. annulliert werden können.

Die Reliabilität kann durch die rechnerunterstützte Ausführung nicht garantiert werden, jedoch kann ihr Koeffizient rechnerunterstützt ermittelt werden (nur bei der Verwendung der gleichen Aufgaben für alle Teilnehmenden). Die Validität kann weder rechnerunterstützt garantiert noch bestimmt werden.

Ausgehend von den erarbeiteten Grundlagen wurden die Anforderungen an eine rechnerunterstützte Testumgebung spezifiziert. Es stellte sich heraus, das ein Werkzeug zur Testdurchführung sowie ein Werkzeug zur Auswertung benötigt werden. Zusätzlich wurden die Anforderungen an ein Werkzeug zur Test- und Aufgabenadministration beschrieben, das die Verwaltung der Datenbank erleichtert.

Anschließend wurde ein geeignetes Datenbankschema entwickelt, in dem sowohl die Aufgaben- und Testdaten als auch die Ergebnisse der Tests verwaltet werden können. Dabei wurde besondere Sorgfalt darauf verwendet, die Konsistenzbedingungen in der Datenbank zu bestimmen, damit keine unvollständigen Tests (z.B. zu wenig Aufgaben) in der Datenbank gespeichert werden können.

Ausgehend von den Anforderungen und dem Datenbankschema wurden ein Werkzeug zur Testdurchführung (WETED) sowie ein Werkzeug zur Bewertung und Auswertung von

Tests (WEBEWAU) entworfen.

Für die entwickelten Werkzeuge wurden die wesentlichen SQL-Anfragen formuliert, die für die Durchführung und Analyse der Tests benötigt werden.

Während der Arbeit wurden bereits praktische Erfahrungen bei der rechnerunterstützten Testdurchführung gesammelt. Es wurden in einer zu dieser Diplomarbeit parallel verlaufenden Veranstaltung rechnerunterstützte Tests ausgeführt. Die Testdurchführung erfolgte noch nicht mit den entworfenen Werkzeugen und die Daten wurden zuerst in einem einfachen Datenbankschema gespeichert. Zur Überprüfung des Datenbankschemas und der SQL-Anfragen wurden die Daten in das entworfenen Datenbankschema gebracht. Dabei zeigte sich, das ein neuer Aufgabentyp beim Entwurf nicht beachtet wurde, die Umfrage-Aufgabe. Mit kleine Abstrichen (alle Items werden als falsch gewertet) konnte auch dieser Aufgabentyp in die Datenbank integriert werden. Die beschriebenen SQL-Anfragen wurden getestet und erfüllten ihre Aufgaben. Die Ergebnisse der Anfragen sind jedoch aufgrund der Stichprobengröße und der geringen Anzahl der Aufgaben pro Test nur bedingt aussagekräftig.

Ausblick

9.0.5 Implementierung

Als nächster Schritt müssen zunächst die in der Arbeit entworfenen Werkzeug zur Testdurchführung und Auswertung vollständig implementiert werden. Die Datenbank ist bereits mit einer kleinen Aufgaben- und Testsammlung vorhanden, so daß die Werkzeuge sofort getestet werden können.

Für den praktischen Einsatz ist es wünschenswert, ein Werkzeug zur Test- und Aufgabenadministration zur Verfügung zu haben, um die Aufgaben und Testdaten nicht von Hand in die Datenbank eingeben zu müssen. Dazu muß das entsprechende Werkzeug entworfen und implementiert werden. Die Anforderungen sind bereits in dieser Arbeit beschrieben.

9.0.6 Rechtliche Situation

Die rechtliche Situation zur Durchführung bewerteter Tests ist zumindest an der Universität Stuttgart noch nicht gegeben. So wird in der Prüfungsordnung ([PO94]) in Paragraph 9 Absatz 1 verlangt, daß Prüfungsleistungen von mindestens einem Professor bewertet werden. Bei einer rechnerunterstützten Bewertung ist dies nicht gegeben. Es kann zwar argumentiert werden, daß die Eingabe der Aufgaben (inkl. der richtigen Lösungen) und der Bewertungsart sowie den entsprechenden Schlüssel durch einen Professor einer Bewertung durch diesen gleichkommt, zur Klärung der rechtlichen Situation muß die Prüfungsordnung aber entsprechend erweitert werden.

Durch verteilte Lehrveranstaltungen werden Studenten Tests von anderen Universitäten

ausführen. Die Anerkennung dieser Leistungen ist bereits in der Prüfungsordnung (Paragraph 15, Absatz 1) geregelt.

9.0.7 Praktischer Einsatz

Es ist gedacht, die in dieser Arbeit entwickelten Werkzeuge in einer verteilten Lehr- und Lernumgebung zum Einsatz zu bringen. Sinnvollerweise sollte diese Umgebung ebenfalls datenbankunterstützt sein und in das in dieser Arbeit entworfenen Datenbankschema integriert werden. In [Ste97] wird der Entwurf einer datenbankgestützten Lehr- und Lernumgebung vorgestellt.

Bevor die Werkzeuge zur Bewertung von Teilnehmenden herangezogen werden, müssen zuerst Erfahrungen bei der Generierung der Aufgaben gesammelt werden. Dies könnte in einem ähnlichen Rahmen wie beim durchgeführten Hauptseminar oder auch bei einer größeren Veranstaltung erfolgen.

Eine Bewertung von Teilnehmenden sollte nach einem ausreichenden positiven Erfahrungsgewinn zunächst bei “unbedeutenden” Prüfungen, wie beispielsweise einer Scheinklausur vorgenommen werden.

Erst bei einem erfolgreichen Einsatz in diesem Gebiet sollten die Werkzeuge für die Benotung von Teilnehmenden verwendet werden.

Literaturverzeichnis

- [Big69] Franz Biglmaier. *Leistungsmessung durch informelle Lehrertests*
In: betrifft: Erziehung, 2. Jg. (1969), Nr. 3, S. 22-26 (1. Teil), Nr 4, S. 26-28 (2. Teil), 1969
- [Cha96] Don Chamberlin. *Using the new DB2 - IBM's Object-Relational Database System*
Morgan Kaufmann Publishers, 1996
- [CoHa94] Carol Cooper, Perry N. Halkitis. *No More Pencils, No More Books...: The Revolution in Computer-Adaptive Testing*
<http://synapse.tricon.net/coop.html>, 1994
- [Duden91] Herausgegeben von der Dudenredaktion. *Duden Band 1 - Die deutsche Rechtschreibung*.
20., neu bearbeitete und erweiterte Auflage, Dudenverlag, 1991
- [Ehe77] W.P. Eheim. *Zur Beeinflussbarkeit der Schwierigkeiten von Mehrfachwahl-Aufgaben*
Diagnostica, 1977
- [GaBe86] Nathaniel L. Gage, David C. Berliner. *Pädagogische Psychologie*
4., völlig neu bearbeitete Auflage, Psychologie Verlags Union Beltz, 1986
- [GrRe93] Jim Grey, Andreas Reuter. *Transaction Processing: Concepts and Techniques*
Morgan Kaufmann Publishers, Inc; 1993
- [Hor71] P. Horst. *Messung und Vorhersage*
Beltz Verlag, 1971
- [HSMMA98] Hauptseminar *Multimediale Anwendungen im Internet WS97/98*
<http://www.informatik.uni-stuttgart.de/ipvr/as/lehre/hauptseminar/HSMMA.html>, 1998
- [LiRa94] Gustav A. Lienert, U. Raatz. *Testaufbau und Testanalyse*
5., völlig neubearbeitete und erweiterte Auflage, Psychologie Verlags Union Beltz, 1994

- [LuBe95] Mary E. Lunz, Betty Bergstrom. *Computerized Adaptive Testing: Tracking candidate response patterns*
Journal of Computing Research, Vol. 13(2), S. 151-162, 1995
- [Lud92] Jochen Ludewig u.a. *Studien-Arbeiten: ein Leitfaden zur Vorbereitung, Durchführung und Betreuung von Studien-, Diplom- und Doktorarbeiten am Beispiel Informatik*
Teubner, 1992
- [MiNo95] Christian Michel, Felix Novak. *Kleines Psychologisches Wörterbuch*
4. Auflage, Verlag Herder, 1995
- [Otto95] Marcus Otto. *Pädagogische Theorien und Modelle für das rechnerunterstützte kooperative Lehren und Lernen.*
Diplomarbeit Nr. 1271, Fakultät Informatik der Universität Stuttgart, 1995
- [PaDa36] D. G. Paterson, J. G. Darley. *Men, women and jobs.*
University of Minnesota Press, 1936
- [PaLeSu94] Jody Paul, Susan B. Lewis, Patricia A. Supon. *Improving Education Through Improved Assessment.*
Proceedings: Frontiers in Education, 24th Annual Conferenc, 1994
<http://apex.cudenver.edu/jody/FIE94.html>
- [PO94] *Prüfungsordnung der Universität Stuttgart für den Diplomstudiengang Informatik.*
Fakultät Informatik der Universität Stuttgart, 1994
- [RaKl83] U. Raatz, C. Klein-Barley. *Ein neuer Ansatz zur Messung der Sprachleistung. Der C-Test: Theorie und Praxis.*
In: R. Horn, K. Ingenkamp, R.S. Jäger (Hrsg.). Tests und Trends 3, S. 107-138, Beltz Verlag, 1983
University of Minnesota Press, 1936
- [Reu93] Andreas Reuter. *Skript zur Vorlesung Informationssysteme / Datenbanken WS93/94.*
Universität Stuttgart, Fakultät Informatik, 1993
- [Sche80] Andreas Schelten. *Grundlagen der Testbeurteilung und Testerstellung.*
Quelle und Meyer, 1980
- [Ste97] Michael Steinmetz. *Entwurf eines Datenbankschemas für Web-basierte Lernumgebungen.*
Diplomarbeit Nr. 1489, Fakultät Informatik der Universität Stuttgart, 1997
- [Uls49] Robert Ulshöfer. *Zur Beurteilung von Reifeprüfungsaufsätzen. Auswertung eines gemeinsamen Versuchs der Deutschlehrer.*
In: Der Deutschunterricht, 1. Jg., (1949), H. 8, S. 84-102, 1949

- [Voß92] A. Voß *Skript zur Vorlesung Energiesysteme I - Band 1: - Grundlagen der Energiewirtschaft - und - Fossile Energieträger -*
Institut für Energiewirtschaft und Rationelle Energieanwendung, Universität Stuttgart, 1992
- [WeGrHe89] Weinert, Graumann, Heckhausen u.a. *Pädagogische Psychologie 1 und 2.*
Funk-Kolleg, 1989
- [Wö68] Herbert Wölker. *Zensuren aus dem Computer - objektivierte Auswertung programmierter Prüfungen.*
Manz Verlag, 1968

Erklärung

Hiermit versichere ich, diese Arbeit
selbständig verfaßt und nur die
angegebenen Quellen benutzt zu haben.

(Wolfgang Mahnke)