



Universität Zürich
Institut für Informatik

Aufgabenarten, Bewertungsmethoden & Spielregeln in mobilen Spielen

Diplomarbeit

vorgelegt von

Marcel Schönbächler

Einsiedeln, SZ

03-705-191

angefertigt am

Institut für Informatik

der Universität Zürich

Prof. Dr. G. Schwabe

betreut durch

Christoph Göth

Abgabe der Arbeit:

10. Januar 2008

Zusammenfassung

In der vorliegenden Diplomarbeit wird das Mobile-Learning-Game-Forschungsprojekt „mExplorer“ zuerst didaktisch aufbereitet in dem die beiden Begriffe „Learning“ und „Game“ genauer analysiert werden. Es wird unter anderem beschrieben wie Lernen und Spielen zusammenhängen und wie Spiele benützt werden können um Lerninhalte zu vermitteln.

Im praktischen Teil dieser Arbeit werden die unterschiedlichen Aufgabenarten erweitert sowie das bereits verwendete Bewertungs- und Punktesystem verbessert: Der Spielleiter kann nun zusätzlich zu den bisherigen Aufgaben auch Ketten-, Kreativ- oder Gruppenaufgaben einsetzen und so das mobile Lernspiel noch attraktiver gestalten.

Mit zwei Benutzertests wurde die Wirksamkeit dieser neuen Module überprüft. Es konnte festgestellt werden, dass vor allem Kreativaufgaben zu einem höheren Spielspass führen, was ein Hauptziel dieser Arbeit war.

Abstract

In this thesis, the Mobile-Learning-Game-research-project "mExplorer" will be analyzed firstly to demonstrate how the two concepts of "Learning" and "Game" interact with each other. It will examine how learning and playing games are related, and how they can be used for learning.

The practical part of this thesis shows the different types of tasks that are expanded, and the improved assessment and scoring system: Additionally to the previous tasks that were implemented, the coordinator of the game can use the so-called chain-, creative- or group tasks which result that using the mExplorer is more appealing then it was before.

With two user tests, the effectiveness of these new modules was tested. It was observed that especially creative tasks lead to more fun, which was a main objective of this work.

Inhaltsverzeichnis

1	Einleitung.....	7
1.1	Aufgabenstellung	7
2	Theoretische Einführung	9
2.1	Learning	9
2.1.1	Begriffsdefinitionen „Lernen“	9
2.1.2	Spielen als Üben von Fertigkeiten	9
2.2	Game.....	10
2.2.1	Faktoren eines Spiels.....	10
2.3	Spielend Lernen.....	11
2.3.1	Arten von interaktivem Lernen	12
2.3.2	Welches Spiel für welchen Lerninhalt	13
2.4	Spielkategorien	14
2.4.1	Schnitzeljagd / Schatzsuche.....	14
2.4.2	Rollenspiele	16
2.4.3	Detektivspiele	20
2.4.4	„Multiple-Choice-Spiele“	24
2.5	Lernprozess.....	25
2.5.1	Aufgabenstellungen.....	26
2.5.2	Korrektur und Bewertung.....	30
2.5.3	Feedback	33
2.5.4	Beispiel einer Lernprozess-Implementation	34
3	Experteninterviews.....	36
3.1	Resultate aus den beiden Interviews	36
3.1.1	Kreativaufgaben.....	36
3.1.2	Zusammenarbeit.....	36
3.1.3	Lösungseingaben.....	37
3.1.4	Bewertung der Lösung.....	37
3.1.5	Zeitpunkt der Bewertung und Feedback	38
3.1.6	Weitere Anforderungen	38
4	Szenarien	40
4.1	Phase 1 - Aufgabenstellung	40
4.1.1	Einzelne Aufgaben	40
4.1.2	Muss- / Soll-Aufgaben	40

4.1.3	Kettenaufgaben.....	41
4.1.4	Postenlauf.....	41
4.2	Phase 2 - Lösen der Aufgabe	42
4.2.1	Einzelteams.....	42
4.2.2	Gruppenaufgaben.....	42
4.2.3	„Task Hunting“	43
4.3	Phase 3 - Korrektur / Bewertung.....	43
4.3.1	Spielleiter.....	43
4.3.2	Andere Spieler	43
4.3.3	Rating Score	44
4.3.4	Nominalskala.....	44
4.3.5	Scoring Rubric.....	45
4.4	Phase 4 - Feedback	45
4.4.1	Rangliste.....	45
4.4.2	Feedback	45
4.5	Bildung der Szenarien.....	46
4.5.1	Beispiel-Szenario.....	46
4.6	Analyse der einzelnen Module	47
4.6.1	Module der Phase 1	47
4.6.2	Module der Phase 2	47
4.6.3	Module der Phase 3	48
4.6.4	Module der Phase 4	48
5	Implementation.....	49
5.1	Modulauswahl / Anforderungsspezifikation	49
5.2	Umsetzung der einzelnen Module.....	50
5.2.1	Aufgabenstellung - Kettenaufgaben.....	50
5.2.2	Aufgabenstellung - Einzelne Aufgaben - Kreativaufgaben.....	54
5.2.3	Lösen der Aufgabe - Gruppenaufgaben	63
5.2.4	Korrektur / Bewertung - Spielleiter	68
5.2.5	Korrektur / Bewertung - Rating Score.....	69
5.2.6	Korrektur / Bewertung - Nominalskala.....	70
5.2.7	Feedback - Rangliste	71
5.2.8	Zusätzliche Funktionen	72
6	Software-Tests	75
6.1	Techniktest.....	75

6.1.1	Aufgabe	75
6.1.2	Durchführung	75
6.1.3	Analyse, Erkenntnisse und durchgeführte Verbesserungen	75
6.2	Benutzertest 1	76
6.2.1	Aufgabe	76
6.2.2	Durchführung	76
6.2.3	Durchgeführte Verbesserungen	76
6.3	Benutzertest 2	77
6.3.1	Aufgabe	77
6.3.2	Durchführung	77
6.3.3	Durchgeführte Verbesserungen	78
6.4	Analyse der Fragebogen	78
6.4.1	Analyse Fragebogen Benutzertest 1	78
6.4.2	Analyse Fragebogen Benutzertest 2	80
6.5	Fazit	81
7	Schlusswort	82
7.1	Persönliche Erfahrungen	82
8	Anhang	83
8.1	Aufgabenstellung	83
8.2	Benutzertests - Instruktionen und Aufgaben	84
8.2.1	Instruktionen	84
8.2.2	Warm-Up-Phase	84
8.2.3	Spielphase	85
8.3	Auswertungen der Fragebogen	88
8.3.1	Ausgewerteter Fragebogen Benutzertest 1	88
8.3.2	Ausgewerteter Fragebogen Benutzertest 2	92
8.3.3	Mit SPSS ausgewerteter Benutzertest 1	96
8.3.4	Mit SPSS ausgewerteter Benutzertest 2	97
8.4	Bibliographie	98
8.4.1	Literaturverzeichnis	98
8.4.2	Abbildungsverzeichnis	100
8.4.3	Tabellenverzeichnis	101

1 Einleitung

Der mExplorer ist ein Mobile-Learning-Game-Forschungsprojekt der Universität Zürich. Die Forschung leitet Christoph Göth, welcher auch mit seiner Diplomarbeit „Prototypische Implementierung einer mobilen Spieleumgebung für den PDA“ (Göth 2003) die Grundlagen für diese Forschungsarbeit legte. Der mExplorer ist eine in der Programmiersprache „Java“ programmierte Client-Server-Anwendung. Während der Server alle Spieldaten und Spielkonfigurationen sammelt und diese den Clients verteilt, ist der Client auf einer mobilen Plattform installiert und stellt alle vom Server erhaltenen Daten und Informationen auf dem Bildschirm dar. Der Datenaustausch erfolgt beim Client über das drahtlose Netzwerk, während der Server per Kabel angeschlossen ist. Ziel dieser Umgebung ist es, den erstsemestrigen Studenten spielerisch die Universität zu zeigen, damit sie einerseits einen geführten Rundgang durch die Gebäude haben und sehen, wo sich was befindet (Sekretariat, Mensa, Kopierautomat, etc.), und andererseits andere Kommilitonen kennenlernen. Der mExplorer ist ein Mehrspieler-Game, welches von den Spielern erfordert miteinander zu interagieren. Es sind kompetitive sowie kooperative Interaktionsmöglichkeiten im System eingebaut (Göth und Froberg, "mExplorer" 2008). Dazu hat jedes Zweier-Team einen PDA¹ auf welchem die Gebäudekarte und die aktuelle Position angezeigt werden. Auf der ganzen Karte sind Aufgaben, sogenannte Tasks, verteilt, welche es zu lösen gilt. Dazu läuft das Team in den Gebäuden der Universität herum und bewegt sich zu den einzelnen Aufgaben hin. Eine mögliche Aufgabe ist beispielsweise herauszufinden, wie viele Kopierautomaten auf der Etage ‚G‘ stationiert sind. Die so erhaltene Zahl muss dann im Gerät eingegeben werden.

Aufgrund der Erkenntnisse der verschiedenen Benutzertests (Göth und Froberg, mExplorer - Arbeitsberichte 2007), welche Christoph Göth mit Dirk Froberg bereits durchführte, wurden mit unterschiedlichsten Diplom- und Bachelorarbeiten neue Funktionen dem mExplorer hinzugefügt. Auch diese Diplomarbeit erweitert die Möglichkeiten des mExplorers: Bis jetzt gab es nur Aufgaben, welche direkt vom Computer ausgewertet werden konnten: Multiple-Choice- und Short-Answer-Aufgaben. Diese Aufgabenarten werden nun im Rahmen dieser Arbeit mit einem zusätzlichen Typ erweitert.

1.1 Aufgabenstellung

Diese Arbeit erweitert die momentan geschlossenen Aufgabentypen. Der Spielleiter kann nun auch kreative und offene Aufgaben stellen. Die Lösungen der einzelnen Teams sollen noch während des Spiels vom Spielleiter beurteilt werden können. Da auch die Punkteverteilung überarbeitet wird, soll es ab jetzt möglich sein am Ende des Spiels eine Rangliste zu erstellen, welche die Teams bei Bedarf bewertet und so die Lehrperson das durchgeführte Spiel auch benoten kann. Die Aufgabenstellung, welche ich erhalten habe, finden Sie am Ende in Anhang 8.1 aufgeführt.

Die vorliegende Arbeit besteht im Wesentlichen aus drei Teilen. Im ersten Teil werden die theoretischen Grundlagen von „Spiel“ und „Lernen“ erörtert sowie andere mobile Lernspiele analysiert. Danach wird der Lernprozess bei Übungen und Klausuren aufgezeigt. Die Resultate aus den Interviews mit Dirk Froberg und Christoph Göth runden den theoretischen Teil ab.

Der zweite Teil geht auf die konkrete Umsetzung der gewonnenen Erkenntnisse ein. Es werden aus den durchgeführten Analysen des ersten Teils Module beschrieben, von welchen eine Auswahl auch konkret implementiert wird. Anschliessend werden die Realisierung der neuen Aufgabentypen und die neu programmierten Codeteile genauer beschrieben.

¹ Personal Digital Assistant: kleiner tragbarer Computer. Auch bekannt unter dem Namen Handheld-Gerät.

Der letzte Teil behandelt die durchgeführten Benutzertests. Es wird zuerst der Ablauf geschildert, gefolgt von den Auswertungen der Fragebogen. Eine Übersicht über die getroffenen Massnahmen und Anpassungen im Code werden am Ende dieses Teils aufgeführt.

Es ist mir ein Anliegen allen zu danken, welche zum Gelingen dieser Arbeit beigetragen haben. In erster Linie ist dies Christoph Göth, welcher bei Problemen für mich da war. Dann die vielen Testpersonen, ohne sie hätte ich die Wirkungen der neuen Funktionen nicht überprüfen können. Besonders bedanken möchte ich mich bei meinen Angehörigen, welche mich immer tatkräftig unterstützten und motivierten.

Einsiedeln, im Januar 2008

Marcel Schönbächler

2 Theoretische Einführung

Der mExplorer ist ein sogenanntes Mobile Learning Game. Dieser Ausdruck enthält die beiden Begriffe „Learning“ und „Game“. Ganz zu Beginn werde ich analysieren, wie Lernen definiert wird und weshalb Spielen und Lernen eng miteinander verknüpft sind. Anschliessend zeige ich auf, welche Faktoren ein „Game“, ein Spiel, ausmachen. Arten von interaktivem Lernen und eine Übersicht über Lerninhalte werden als Zusammenführung zwischen den beiden Teilbereichen genauer erläutert. Nach einer Analyse von mobilen Spielen folgt ein theoretischer Teil eines Lernprozessablaufs. Dort zeige ich auf, wie eine Interaktion zwischen Lehrperson und Lernendem funktionieren kann. Dies bildet eine wichtige Basis für die nachfolgende praktische Implementation von Kreativaufgaben.

2.1 Learning

2.1.1 Begriffsdefinitionen „Lernen“

Viele Wissenschaftler haben das Lernen untersucht und auch häufig Definitionen zum Begriff „Lernen“ entwickelt. Ich führe an dieser Stelle ein paar interessante Auslegungen an und werde am Ende einen kurzen Kommentar zu diesen geben:

"Unter Lernen verstehen wir den Erwerb, die Veränderung oder den Abbau von Erlebens- und Verhaltensweisen durch bestimmte Umwelterfahrungen." (Schmitt 1999, S.1)

„Lernen umfasst alle Verhaltensänderungen, die aufgrund von Erfahrungen zustande kommen.“ (Lefrancois 1994, S. 3)

"Lernen [kann man] als einen Prozess definieren, der zu relativ stabilen Veränderungen im Verhalten oder im Verhaltenspotential führt und auf Erfahrung aufbaut." (Zimbardo 1992, S. 227)

„Lernen umfasst: Auffassen, Verstehen, Einsicht in Zusammenhänge gewinnen, Klarheit und Durchblick erwerben, Speichern und Abrufen von Wissen sowie das Verknüpfen mit Anwendungen.“ (Schweizer Armee 2005, S. 4)

Bei allen Definitionen des Lernens sind „Veränderungen“ elementar. Nur wenn der Lernende sein Verhalten ändert, darf man von Lernen sprechen. Diese Verhaltensänderungen wie es Lefrancois und Zimbardo nennen, sind die Resultate der Lernprozesse.

2.1.2 Spielen als Üben von Fertigkeiten

Während ein Kind spielt, verändert es automatisch seine Fähigkeiten und sein Können. Diese kann es in verschiedensten Situationen wieder verwenden. „Durch Spielerfahrungen erweitern Kinder auch ihr Wissen von dieser Welt und deren Bedingungen. (...) Ein erheblicher Vorteil des Spielens besteht darin, dass das Kind dabei etwas ausprobieren kann, ohne dass es das [unangenehme] Erlebnis des Versagens befürchten muss.“ (Mietzel 2002, S. 158).

Die folgende Liste über die Vorteile der Verwendung von Spielen zum Lernen wurde dem Buch „The Primary English Teacher’s Guide von Jean Brewster et al. (S. 173) entnommen:

- Spiele erhöhen die Anzahl unterschiedlicher Lernsituationen.
- Spiele wechseln den Rhythmus innerhalb der Lektion und helfen dem Schüler die Motivation aufrechtzuerhalten.
- Spiele lockern das eher formelle Lernen auf und helfen die „Energie“ des Schülers zu erneuern.
- Spiele können helfen die Aufmerksamkeit, die Konzentration und das Gedächtnis zu verbessern.

- Lernende werden zum Teilnehmen angespornt. Auch Schüchterne können motiviert werden mitzumachen.
- Spiele helfen eine Spass-Atmosphäre zu erzeugen und reduzieren die Distanz zwischen Lehrer und Schüler.
- Spiele können Schwachpunkte auf eine angenehme Weise aufzeigen und so das Bedürfnis für weiteres Training wecken.

„Kenntnis verfestigt sich erst durch Übung.“ Dieses Zitat von Monika Bütler zeigt indirekt den gemeinsamen Nenner von Spielen und Lernen auf: Übung. Spielt ein Kind gerne das Rechenspiel am Computer, übt es immer und immer wieder mit den Zahlen zu jonglieren. Es spielt sehr oft auch in der Freizeit dieses Spiel. Es trainiert so komplett unbewusst die Rechenoperationen, welche es in der Schule gelernt hat. Durch das häufige Spielen werden die Fertigkeiten geübt und gefestigt. Diese gehen in „Fleisch und Blut“ über und das Kind kann jederzeit auf diese zurückgreifen. Dieses Kenntnis hat sich so verfestigt. Das Kind hat die Rechenoperationen gelernt.

2.2 Game

Die folgenden Ausführungen behandeln das Thema „Spiel“. Sie zeigen auf was für Anforderungen erfüllt sein müssen, damit ein Spiel auch wirklich mit „Leib und Seele“ gespielt wird. Zuerst werde ich diese Faktoren, welche ein Spiel ausmachen, aufzeigen und dann verschiedene Spiele analysieren.

2.2.1 Faktoren eines Spiels

Marc Prensky führt in seinem Buch sechs Faktoren auf, welche ein Spiel ein Spiel werden lassen:

1. Regeln

Jedes Spiel basiert auf gewissen Regeln. Diese sorgen dafür, dass ein Spieler innerhalb gewisser Leitplanken agiert um ein definiertes Ziel zu erreichen. Auch sind Regeln dazu da um allen Spielern zu versichern, dass jeder Spieler denselben Pfad nimmt. Sie definieren also die Grenzen was in Ordnung ist und was nicht; was fair und was unfair ist im Spiel.

Damit Regeln aber auch durchgesetzt werden können, gibt es in jedem Spiel einen Schiedsrichter, welcher für die Einhaltung der Regeln sorgt. Bei einem Kartenspiel unter Kollegen können dies Kameraden sein, welche einen Betrüger aus dem Spiel ausschliessen. Beim Fussball ist es der Schiedsrichter, welcher mit gelben und roten Karten die Spieler zu Fairness „zwingt“. Bei Computerspielen werden die Regeln von den Programmierern in das Spiel eingebettet, damit der Benutzer gar keine Chance hat sich nicht regelkonform zu verhalten.

2. Ziele

Ziele sind wichtig, weil der Mensch eine ziel-orientierte Spezies ist. Die Zielerreichung ist einer der grossen Faktoren, welche die Spieler motivieren immer wieder zu spielen. Der Spieler misst sich selber ob er ein vorgegebenes Ziel erreichen kann oder nicht. Einzelne Spielentwickler gehen gar so weit und sagen, dass ein Spiel ohne Ziele lediglich ein Spielzeug sei, mit dem man so spielen kann, wie man gerade möchte. Ziele stossen uns an etwas erreichen und gewinnen zu wollen.

3. Ergebnis und Feedback

Jedes Spiel hat am Ende ein Ergebnis und gibt dem Spieler durch dieses Resultat ein Feedback: Bei klassischen Spielen kann es sein, dass man gegen den Gegner gewonnen oder verloren hat. Gewinnen und Verlieren haben starke emotionale und „ego-befriedigende“ Auswirkungen, was einen grossen Anteil der Anziehungskraft von Spielen ausmacht.

Das Feedback, welches wir vom Spiel erhalten, gibt uns immer sofort die Information ob es für uns positiv oder negativ aussieht; ob wir innerhalb der Regeln spielen oder möglicherweise die Re-

geln gebrochen haben („Rote Karte“); ob wir uns dem Ziel nähern oder entfernen („warm oder kalt“) sowie ob unsere Leistung reicht für einen Eintrag in der High-Score-Tabelle oder nicht. Feedback kann einerseits ein numerischer Wert sein oder auch in anderer Form aufbereitet werden (zum Beispiel grafisch; mündlich).

Findet im Spiel ein Lerneffekt statt, so wird dieser durch das Feedback erzeugt. Das Feedback ist dafür zuständig, dass ein Spieler aus seinem Verhalten die (richtigen) Konsequenzen zieht und zukünftig anders agieren wird.

Die Art wie Feedback dargestellt wird, ist enorm wichtig und komplex: zu viel oder zu wenig führen sehr schnell zur Frustration des Spielers. Daher ist es notwendig, dass sich die Schwierigkeit des Spiels dem Spieler anpasst. Um einem Einsteiger das Spiel zu erleichtern, sollte er rasche Erfolge erzielen können. Bei einem Profi hingegen ist es notwendig, dass das Spiel anspruchsvoll genug ist, dass er nicht die Freude daran verliert und so die Herausforderung bestehen bleibt das Spiel weiterhin zu spielen.

4. Wettbewerb und Herausforderung

Mit Wettbewerb und Herausforderung sind alle Probleme eines Spiels betroffen, welche der Spieler zu lösen hat. Diese zwei Faktoren sind das, was das Adrenalin und die Kreativität fördern sowie die Begeisterung für das Spiel auslösen. Ein Problem lösen zu müssen („wie setze ich diese 6‘000 Teile des Puzzles zusammen?“; „wie erziele ich den höchsten Wert in der High Score Tabelle?“) ist zentral damit der Spieler Freude am Spiel hat und immer wieder spielt.

5. Interaktionen

Einerseits gibt es die Interaktionen zwischen dem Spieler und dem Gegenüber (Computer, anderer Spieler), welche beim Punkt „Feedback“ schon diskutiert wurden. Andererseits gibt es bei Spielen immer auch einen sozialen Aspekt zu berücksichtigen. Man spielt mit anderen Personen, was die Formierung von sozialen Gruppierungen fördert. Während man auch alleine spielen kann, macht es meistens mehr Spass zusammen zu spielen. Dies ist der Hauptgrund weshalb heute sogenannte Multiplayer-Spiele wie World of Warcraft² oder Counter-Strike³ so hohen Zulauf geniessen. Spieler ziehen normalerweise menschliche Gegner vor und haben so mehr soziale Interaktionen auch wenn nicht zwingend Face-to-Face-Kontakte.

6. Repräsentation und Thema

Repräsentation bedeutet, dass ein Spiel sich um ein spezielles Themengebiet dreht: Während es bei „Tetris“ um Musterbildung und -erkennung geht, dreht sich „The Age of Empire“⁴ um das Krieg führen in der Zeit der alten Römer.

Repräsentation enthält auch das Element „Fantasie“. Die Fantasie kann sich um den Weltraum, um das Mittelalter oder sonstige Science-Fiction-mässige Inhalte drehen. Den Spielentwicklern sind keine Grenzen gesetzt. Die Story und die Erzählung in Spielen werden immer wichtiger, obwohl die Integration der beiden weder leicht noch offensichtlich ist, aber dennoch von den Konsumenten erwartet wird.

2.3 Spielend Lernen

Es gibt viele Arten wie spielend gelernt werden kann. Zum Einstieg dieses Teils werde ich zuerst auf eine Auswahl möglicher Schemata von interaktivem Lernen eingehen. Nur wenn das Lernen interaktiv erfolgt,

² <http://www.wow-europe.com/> (08. August 2007)

³ http://www.electronic-arts.de/publish/lea_spiele/spiele_info.html?etitel=Counter-Strike%99%3A+Source%99 (08. August 2007)

⁴ <http://www.microsoft.com/games/empires/> (08. August 2007)

ist es auch möglich, dass das Lernen zum Spiel wird. „Das Gegenüber“ wird zum Verbündeten mit welchem man ein Ziel erreichen muss, oder zum Gegner, den es zu schlagen gilt.

Als zweite Stufe zeige ich anhand einer Aufstellung welcher Lerninhalt mit welchen Spielstilen am besten vermittelt werden kann, Denn nicht jedes Spiel ist für ein gegebenes Lernziel optimal.

2.3.1 Arten von interaktivem Lernen

Die folgenden Ausführungen sind wiederum an Marc Prensky's Digital Game-Based Learning (S. 145ff) angelehnt.

- Anwendung und Bewertung

Diese Form ist eines der ältesten Lernschemata bei welchem Computer eingesetzt werden: Der Computer zeigt ein Problem an (zum Beispiel: was ergibt 5x5?) und der Lernende muss sein gelerntes Wissen *anwenden* und das korrekte Rechenresultat eingeben. Der Computer *bewertet* das Resultat auf seine Richtigkeit.

Diese Art von interaktivem Lernen ist eine hervorragende Möglichkeit Lerninhalte zu üben, welche viel Repetition benötigen wie zum Beispiel das Lernen von Fremdsprachenwörtern.

- Learning by Doing

Etwas lernen in dem man die Lerninhalte direkt umsetzt und verwendet. Der Lernende wird durch die häufige Repetition immer routinierter und beherrscht je länger je mehr die vorgegebenen Inhalte.

- Von Fehlern lernen (Feedback Coaching)

Wenn der Lernende einen Fehler macht, erhält er ein Feedback, welches ihm aufzeigt, was er hätte anders machen müssen, um das vorgegebene Ziel zu erreichen. Dies ist genau das, was in (Computer-) Spielen angewendet wird: Bei einer Rennsimulation schafft es der Spieler nicht in der vorgegebenen Zeit eine gewisse Strecke zu fahren. Je öfter er diese Aufgabe wiederholt, desto mehr kennt er die Rennpiste und weiss, wo noch einzelne Sekundenbruchteile zu gewinnen sind. Das Feedback ist also, dass der Spieler verliert, wenn die Zeit abgelaufen ist, wenn die Spielfigur gestorben ist oder der Gegner die Armee zerstört hat, etc.

- Zielorientiertes Lernen

Viele Lernspiel-Designer sind der Ansicht, dass es nicht relevant ist zu wissen was das Ziel ist, sondern dass der Lernende weiss, wie das Ziel erreicht wird. Wie oben aufgezeigt wurde, ist das Ziel der Schlüssel jedes Spiels. Es ist das was ein Spiel ausmacht.

- Aufgabenbasiertes Lernen

Aufgabenbasiertes Lernen ist eine Variante von Learning by Doing. Während bei Learning by Doing noch zuerst die Theorie vorgelagert wird, ist diese beim aufgabenbasierten Lernen stark verkleinert. Die Aufgaben sind daher so gestellt, dass sie aufeinander aufbauen und nahezu die gesamte Theorie beinhalten. Der Lernende wird klar durch den Stoff geführt und nicht mit einem Berg von Lernmaterial zu geschüttet in dem er sich zuerst zurecht finden muss. Die potentielle Gefahr bei diesem Ansatz ist, dass der Schüler zu wenig die Theorie hinter den Aufgaben lernt. Es ist also notwendig, dass die Lehrperson auch kreative Wege berücksichtigt, damit die theoretische Basis beim Lernenden verankert werden kann.

- Rollenspiel

Rollenspiele werden oft dazu verwendet, um sogenannte „Soft Skills“ wie Interviewing, Kommunikationstraining, Verkauf und ähnliches zu trainieren. Im Gegensatz zu interaktiven Trainings, welche häufig sehr strukturiert und auch kürzer sind, dauern PC-Rollenspiele häufig mehrere

Stunden, Tage oder sind sogar ohne Ende. Erhöht man die Dauer solcher Rollenspiele erhöht dies ebenso das Lernen des Schülers.

- Intelligente Tutoren

Ein intelligenter Tutor untersucht die Antworten des Schülers und versucht herauszufinden, warum die einzelnen Fehler gemacht wurden. Aufgrund dieser Analyse ist es dann möglich, ein auf den Lernenden zugeschnittenes Feedback zu geben. Das Ganze basiert auf einem Computermode, welches von Experten entwickelt wurde, welche Schüler beim Lösen beobachteten und so Regelmäßigkeiten in den falschen Antworten entdecken konnten. So kennen die Rechner viele häufige Fehler und können nützliche Hinweise geben. Die Zukunft wird dahin gehen, dass der Computer versucht herauszufinden, wie der Spieler denkt und vorgeht (alles anhand seiner Antworten), und er wird aufgrund dieser Analyse das Spiel, seine Strategien und Optionen anpassen.

2.3.2 Welches Spiel für welchen Lerninhalt

Die nachfolgende Tabelle gibt eine Übersicht für welchen Lerninhalt welche Spielarten geeignet sind. Die Spalte „Lernaktivität“ gibt an wie der Lerninhalt trainiert werden kann. Die Lernaktivität sollte also im Spiel enthalten sein, damit der Lerninhalt gefestigt wird. Diese Tabelle wurde von Marc Prensky's Digital Game-based Learning (S. 156) entnommen.

Lerninhalt	Beispiele	Lernaktivitäten	Geeignete Spielarten
Fakten	Gesetze, Produktspezifikationen	Fragen Erinnern Verbinden Drill	Wettbewerb Lernkarten Memory Sportspiele
Fähigkeiten	Interviewing, Verkaufstraining, Projektmanagement	Imitieren, Feedback Coaching, ständige Wiederholung mit ansteigendem Schwierigkeitsgrad	Rollenspiele Abenteuerspiele Detektivspiele
Urteilsvermögen	Managemententscheidungen, Ethik	Fälle analysieren, Fragen, Entscheidungen fällen, Feedback, Coaching	Rollenspiele Detektivspiele Mehrspielerinteraktionen Abenteuerspiele Strategiespiele
Theorien	Marketingstrategien Geldpolitik etc.	Logik Experimentieren Fragen	Open-End-Simulationen Entwicklungsspiele Konstruktionsspiele
Schlussfolgerungen	Strategisches und taktisches Denken, Qualitätsanalyse	Probleme Beispiele	Puzzles
Prozesse / Prozeduren	Auditing, Strategieentwicklung	Systemanalyse und Mikrobetrachtung Anwendung der Prozesse	Strategiespiele Abenteuerspiele Simulationsspiele
Kreativität	Erfindungen Produktgestaltung	Spielen Memory	Puzzles Erfindungsspiele
Sprache	Akronyme Fremdsprachen Berufsjargon	Imitieren wiederholtes Anwenden	Rollenspiele Reflex-Spiele Lernkarten-Spiele

Systeme	Gesundheitsvorsorge Finanzmarkt	Verstehen der Grundlagen, abgestufte Aufgaben, Spielen in Mikrowelten	Simulationsspiele
Beobachtung Feststellung	Haltung, Probleme, Ineffizienz	Beobachten Feedback	Konzentrationsspiele Abenteuerspiele
Kommunikation	Angemessene Sprache, Timing, Beteiligung	Imitierung Anwendung	Rollenspiele Reflex-Spiele

Tabelle 1: Lerninhalt, Lernaktivität und geeignete Spielarten

2.4 Spielkategorien

In diesem Kapitel werden vier Arten von Spiele analysiert, in denen es mobile Spiele gibt. Die Analyse ist wie folgt gegliedert: zuerst beschreibe ich die Spielart, dann analysiere ich ein bis zwei bereits umgesetzte Spiele anhand der oben erwähnten Faktoren. Ich zeige auf, wie die sechs Spielfaktoren umgesetzt sind und welche Lerninhalte im vorliegenden Spiel geübt und gefestigt werden.

2.4.1 Schnitzeljagd / Schatzsuche

Im Internet gibt es unzählige Varianten von Schnitzeljagden zu finden. Ich habe mich hier auf eine beschränkt:

Ein Organisator versteckt einen kleinen Schatz an einem beliebigen Ort seiner Wahl. Die Schatzsucher müssen vom Startpunkt aus diesen Schatz finden. Am Start erhalten sie eine Karte und eine Aufgabe, welche sie lösen müssen, um die nächsten Koordinaten zum Schatz zu erhalten. Bei diesem Wegpunkt haben sie wieder eine Aufgabe zu lösen und erhalten wieder einen neuen Hinweis. Das Spiel beinhaltet diverse Aufgaben, welche die Schatzsucher immer näher zum Ziel, zum Schatz, bringen. Gewonnen haben die Schatzsucher, welche den Schatz als erstes gefunden haben. Als Beispiel für eine solche Schnitzeljagd habe ich Geo-Caching ausgewählt.

Geo-Caching

Geo-Caching⁵ ist die moderne Form von Schnitzeljagd. Das Prinzip ist dasselbe wie das oben beschriebene Spiel. Nur erhält der Spieler keine Karte, sondern gibt die Koordinaten der nächsten Aufgabe in sein GPS-Gerät ein. Dieses GPS-Gerät zeigt dann die Richtung an, in der sich der nächste Hinweis befindet. Durch das Lösen von Aufgaben an den jeweiligen Koordinaten erhalten die Spieler immer wieder neue Punkte mit neuen Hinweisen und nähern sich so dem gesuchten Schatz. Diese Art von Geo-Caching wird „Multi-Cache“ genannt (Ammann 2006). Hat man den Schatz gefunden, muss man sich im Log-Buch eintragen und den Schatz mit einem gleichwertigen Objekt tauschen. Man nimmt also den Inhalt der Schatzkiste mit und legt dafür ein Objekt in die Schatztruhe, welches mindestens denselben Wert hat. Dadurch haben auch nachfolgende Schatzsucher noch ein Erfolgserlebnis wenn sie den Schatz gefunden haben.

Im Moment ist ein grosser Geo-Caching-Boom auszumachen: Es gibt weltweit über 400'000 solcher Geo-Caches. Die Webseite www.geocaching.com sammelt die Koordinaten solcher Verstecke. Für Geo-Caching in der Schweiz wird die Webseite www.swissgeocache.ch empfohlen. Auf dem unten abgebildeten Grafikausschnitt stehen alle roten Punkte für einen einzelnen Cache. Diese Grafik soll einen Eindruck hinterlassen wie weit verbreitet dieses Spiel auch in unseren Regionen ist.

⁵ Cache: engl. geheime Vorratskammer



Abbildung 1: Ausschnitt Geo-Caching Karte "Nordost-Schweiz"⁶

Analyse von Geo-Caching anhand der sechs Spielfaktoren

1. Regeln

Die vorhandenen Regeln sind nicht so ausgestaltet, dass sie auch bei Betrügern durchgesetzt werden können. Ich würde sie eher als eine Art „Geo-Caching-Knigge“ bezeichnen. Ein Beispiel für eine solche Anweisung folgt hier: Der Spieler sollte darauf achten, dass er unbeobachtet von „Geomuggels“⁷ ist, wenn er einen Schatz aushebt. Es ist immer möglich, dass Unkundige einen Cache versehentlich oder mutwillig zerstören. Zusätzlich sollte er sich ins Logbuch mit Datum, Zeit und Name eintragen, damit es für den Eigentümer des Caches spannend ist nachzuvollziehen wer alles seinen Schatz gefunden hat.

2. Ziel

Das Ziel des Spiels ist den Schatz zu finden. Das Objekt zum Tauschen, welches sich im Schatz befindet, ist eher zweitrangig, geht es doch mehr um das Finden des Caches.

3. Feedback und Ergebnis

Das Feedback für den Spieler ist, ob er den Cache gefunden hat oder nicht. Weiter ist auch das Eintragen im Logbuch ein Feedback: je weniger Einträge es hat, desto weniger Personen haben den Cache schon besucht, desto „exklusiver“ ist der Cache.

Feedback gibt auch das GPS-Gerät in dem es anzeigt, ob man sich in die richtige Richtung bewegt oder ein Richtungswechsel notwendig ist.

4. Wettbewerb und Herausforderung

Die Herausforderung beim Geo-Caching ist, ob man die Aufgaben lösen kann und den Schatz findet. Es ist quasi ein Spiel zwischen dem, der den Schatz versteckt, und dem, der ihn sucht. Der eine will den Schatz möglichst gut tarnen, damit er praktisch nicht gefunden wird, während der andere ihn möglichst schnell finden will.

5. Interaktionen

Die Interaktionen mit anderen Cachern finden hauptsächlich über das Internet statt. Über die beiden obenerwähnten Internetseiten entstand eine grosse Community. Mit Foren und Online-Log-Einträgen über gefundene Caches informiert man andere Sucher sowie den Besitzer des Schatzes, dass der Cache nicht von Unkundigen weggeräumt oder zerstört wurde, sondern sich immer noch an seinem ursprünglichen Platz befindet.

6. Repräsentation und Thema

Das Thema von Geo-Caching ist hauptsächlich, dass der Sucher sich draussen in der Natur aufhält. Häufig wird der Schatz noch so versteckt, dass er in der Nähe eines Aussichtspunktes ist (vor allem in den ländlicheren Regionen). Dem Schatzsucher wird so noch einen tollen Ausblick auf

⁶ http://www.swissgeocache.ch/map/showmap_ne.php (08.August 2007)

⁷ Ein Nicht-Geocacher. Damit sind Menschen gemeint, welche Geo-Caching nicht kennen. (Ammann 2006)

die Umgebung ermöglicht. Zusätzlich ist ein weiteres Thema das Üben des präzisen Navigierens mit GPS-Geräten.

Spiel und Lernen

Geo-Caching ist ein Detektivspiel bei dem es darum geht einen versteckten Schatz zu finden. Es werden folgende Lerninhalte eingesetzt:

- Fähigkeiten
Durch das Navigieren per GPS trainiert der Spieler Elemente aus der Karte in der Realität zu erkennen (zum Beispiel eine Wegkreuzung) und reale Elemente auf der Karte zu finden. Geo-Caching trainiert so das Lesen von Landkarten.
- Urteilsvermögen, Kommunikation
Es müssen Entscheidungen im Team gefällt werden, ob man zum Beispiel den rechten oder linken Weg geht. So erzwingt Geo-Caching miteinander zu kommunizieren und sich gegenüber den anderen Teammitgliedern durchzusetzen.
- Kreativität
Die Kreativität des Spielers wird dadurch gefördert, dass er einen Schatz suchen muss. Ist er beim Punkt angelangt, wo der Cache versteckt ist, muss er die Umgebung genau durchforsten. Einerseits ist das GPS nicht auf den Meter präzise und andererseits sind die Schätze jeweils gut versteckt, um sie zu schützen, damit Geomuggels sie nicht finden.

2.4.2 Rollenspiele

Als Rollenspiel bezeichnet man ein Spiel für eine bestimmte Anzahl Personen. Diese Menschen verwenden ihre Vorstellungskraft um eine andere Person, einen fiktiven Charakter mit einem meist aufregenderen Leben darzustellen (Deutsch 1997). Sie benutzen die eigene Fantasie und meist ein Regelwerk, welches Strukturen und Eingrenzungen des Spiels vorgibt. Diese Rollenspiele können in einer Fantasie-Welt (mit Elfen und Zwerge) stattfinden oder die reale Welt als Hintergrund haben. Ziel solcher Rollenspiele ist es häufig, dass der Spieler neue Erfahrungen sammelt, welche er auch in der realen Welt verwenden kann (Wikipedia 2004).

ARCHIE

Archie ist eine PDA-Computer-Software, welche im gallo-römischen Museum in Tongeren (Belgien) einen Besucher durchs Museum führt und ihm dynamisch Informationen präsentiert. Das Ziel von Archie ist auf die verschiedenen Bedürfnisse der Besucher(-gruppen) einzugehen und auch Interaktionen zwischen den Einzelnen zu fördern. So haben die Besucher die Möglichkeit über Voice-over-IP (VoIP)⁸ miteinander zu kommunizieren oder die indirekte Kommunikationsmöglichkeit des Datenaustausches zu wählen. Um die Erwartungen und Bedürfnisse von (Schul-) Gruppen besser zu erfüllen und eine kollaborative Lernaktivität zu erzeugen, wurde ein Museumsrollenspiel programmiert.

Es gibt drei Farmer und einen Leader. Jede Rolle hat seine eigenen Aufgaben und Ziele, welche erreicht werden müssen. Die Farmer haben Fragen zu den Ausstellungsobjekten im Museum zu lösen und erhalten für jede richtige Antwort austauschbare Güter wie Rinder und Schafe.

Der Leader hat einen Überblick über alle verdienten, tauschbaren Güter der Farmer und kann diesen Handelsvorschläge unterbreiten. Nur der Leader hat exklusive Kontakte zum Handelsnetzwerk. Er hat die

⁸Voice-over-IP ist die Telefonie über Computernetzwerke oder das Internet. (<http://www.3cx.de/voip-sip/voip-definition.php> (02. Oktober 2007))

Möglichkeit in den Süden zu reisen und dort die Produkte der Farmer gegen Salz und Eisen zu tauschen. Während jeder Farmer das Ziel hat, eine gewisse Höhe an Salz- und Eisenbestände zu halten, ist das Ziel des Leaders zu schauen, dass jeder Farmer zufrieden ist und dadurch ein bronzenes Schwert erhält (Luyten). Das Spiel ist so konzipiert, dass jeder Spieler von den Aktionen der anderen abhängig ist (Van Loon 2007).

Analyse von Archie anhand der sechs Spielfaktoren

1. Regeln

Spielregeln sind vorhanden, welche direkt im Spiel eingebettet sind. So müssen zuerst Fragen gelöst werden bevor der Farmer neue Güter erhält, welche er tauschen kann.

2. Ziele

Jede Rolle hat ein eigenes Ziel, welches erreicht werden soll. Das Lösen von möglichst vielen Fragen um die dadurch erhaltenen Güter in einen möglichst hohen Salz- und Eisenbestand zu tauschen, motiviert die Spieler. Zusätzliche Motivation ist sicherlich die Konkurrenzsituation: Der Spieler muss zwar mit den anderen zusammenarbeiten, aber er will trotzdem der Beste sein und die höchsten Bestände aller haben.

3. Ergebnis und Feedback

Das Ergebnis ist, ob der Spieler die vom Spiel definierten hohen Bestände an Eisen und Salz erreicht hat oder eben nicht (Kriterium: erfüllt oder nicht erfüllt). Als Feedback ist aufzuführen, dass der Spieler immer gerade den aktuellen Bestand seiner Ressourcen sieht und so ausrechnen kann, wie viele Fragen er noch zu beantworten hat, damit er genügend Güter zum Tauschen besitzt.

4. Wettbewerb und Herausforderung

Die Herausforderungen bei Archie sind einerseits das Lösen der Aufgaben und andererseits dass der Leader den Farmern auch die richtigen Angebote unterbreitet und so den Farmern ihre Zielerreichung ermöglicht.

Das Lösen der Aufgaben ist anspruchsvoll, müssen sich doch die Spieler intensiv mit den einzelnen Ausstellungsobjekten des Museums beschäftigen um die Fragestellungen lösen zu können. Hier sind kollaborative Interaktionen zwischen den einzelnen Spielern denkbar um Hinweise zu den Fragen auszutauschen.

5. Interaktionen

Die Interaktionen mit den anderen Spielern finden während dem Spiel über die integrierten Kommunikationskanäle wie VoIP statt. Es ist anzunehmen, dass auch nach dem Ende des Spiels die einzelnen Teilnehmer miteinander über das Spiel und die einzelnen Ausstellungsobjekte diskutieren und so die Erlebnisse verarbeiten.

6. Repräsentation und Thema

Das Thema dieses Rollenspiels ist das unbewusste Lernen über die ungleichen Schichten in der Gesellschaft, wie diese zustande kamen, wie diese funktionierten und genauer, wie es sich angefühlt haben könnte in früheren Jahrhunderten in diesen ungleichen Positionen zu leben (Luyten). Die Ungleichheit der Klassen wird hier eindrücklich nachgespielt. Die Farmer sind komplett vom Leader abhängig und können ohne ihn ihr Ziel nie erreichen.

Als weiteres Thema sind die Ausstellungsobjekte zu erwähnen: Da die Spieler Fragen beantworten müssen um neue Tauschgüter zu erhalten, haben sie sich zwingend mit der Museumsausstellung zu beschäftigen, wollen sie doch ihr Rollenspielziel erreichen.

Spiele und Lernen

Archie ist eine mögliche Variante eines (einfachen) Rollen- und Simulationsspiels mit dem PDA. Das Hauptziel eines Museumsbesuchs ist normalerweise das Verständnis für die einzelnen ausgestellten Objekte zu gewinnen (Lerninhalt: Fakten). Dies kann mit diesem Spiel erreicht werden. Es sind aber noch weitere Lerninhalte auszumachen:

- Urteilsvermögen, Schlussfolgerungen
Durch die Entscheidungen, welche der Leader und die Farmer zu treffen haben, wird das Urteilsvermögen „was ist fair, was ist unfair“ geschult. Dadurch können die Spieler schlussfolgern, wie das Wirtschaftsleben im Mittelalter gewesen sein könnte, als die Bauern von den Fürsten nach Belieben unterdrückt wurden.
- Systeme, Theorien
Das Prinzip der Tauschwirtschaft, früher Ware gegen Ware, heute Ware gegen Geld, wird hier in einer anderen Form geschult. Auch haben die Schüler die Möglichkeit zu experimentieren im Stile von „was passiert wenn“ ohne dass ihr reales Leben davon Schaden nehmen kann.
- Kommunikation
Da die Spieler untereinander Güter tauschen müssen, ist es notwendig sich abzusprechen. Dies wird in Archie mit einer VoIP-Verbindung realisiert. So wird die Kommunikations- und Teamfähigkeit der einzelnen Spieler trainiert.

CIO Business Game

„Das computer-unterstützte Planspiel ‚CIO Business Game‘ (...) behandelt besonders die vielschichtige Beziehung von IT-Entscheidungen mit allgemeiner Unternehmensführung.

Die Teilnehmer erhalten im Planspiel die Möglichkeit die Geschicke einer Autobank, die eine hohe IT-Durchdringung besitzt, als CIO, CFO, CMO und COO⁹ über mehrere gespielte Jahre zu lenken. Durch kurze Feedbackzyklen werden Fehlentscheidungen aber auch erfolgreiches Handeln offensichtlich und dienen in einem didaktischen Rahmenkonzept als Ausgangspunkt für weitere Diskussionen.“ (Lehrstuhl für Wirtschaftsinformatik)

Das Ziel des Spiels ist mit der Autobank einen möglichst hohen Gewinn zu erzielen. Damit dieses Ziel erreicht werden kann, müssen die vier „Chiefs“ sich koordinieren, damit sie keine zu hohen Ausgaben und demotivierte Mitarbeiter haben sowie viele neue Kunden gewinnen können. In diesem Spiel gibt es sehr viele Parameter, welche zu beeinflussen sind: Mitarbeiterzahl und -lohn, Informatikmittel, Kredite und Darlehen, Preise der Dienstleistungen, etc. Mit der Durchführung von Projekten kann sich die Bank zusätzlich gewisse Fähigkeiten aneignen um noch höhere Gewinne zu erzielen.

Dieses Spiel ist nicht im eigentlichen Sinne ein selbstständiges Programm sondern kann über jeden beliebigen Internet-Browser gespielt werden. Daher ist es auch möglich unterwegs über den internetfähigen PDA die Eingaben ins System zu tätigen.

⁹ CIO: Chief Information Officer; CFO: Chief Financial Officer; CMO: Chief Marketing Officer; COO: Chief Operation Officer

Analyse des CIO Business Game anhand der sechs Spielfaktoren

1. Regeln

Es sind diverse Regeln im Spiel implementiert: So sind zum Beispiel einzelne Projekte von anderen Projekten abhängig und können somit erst ausgeführt werden, wenn das Basisprojekt abgeschlossen wurde. Alle Regeln sind im Spiel integriert und kontrollieren so den Spieler, ob alle seine Handlungen regelkonform sind. Nicht konforme Aktionen werden gar nicht erst zugelassen.

2. Ziele

Das Ziel ist einen möglichst hohen Gewinn zu erzielen. Um dieses Ziel zu erreichen, müssen die Spieler sich absprechen und ihre Tätigkeiten miteinander koordinieren. Machen mehrere Gruppen gleichzeitig am Spiel mit, wird es interessanter für alle. Es sind mehr Gruppen zum Vergleichen vorhanden und die Motivation „die Besten“ zu sein steigt stark an.

3. Ergebnis und Feedback

Nach jeder Runde werden die Finanzaufgaben bekannt gegeben. So wird den Spielern mitgeteilt, ob sie mehr Gewinn oder grössere Verluste in der letzten Periode einführen. Dies bestätigt oder verneint den bisher eingeschlagenen Weg. Ist ein grosser Verlust zu verzeichnen, erzeugt das eine gewisse „Wut im Bauch“ und die Spieler unternehmen alles um wieder in die schwarzen Zahlen zu gelangen. Konnten positive Zahlen vorgelegt werden, motiviert das die Spieler den Gewinn noch weiter zu vergrössern. Das Ergebnis des Spiels ist die Position des Eintrags in der High-Score-Liste.

4. Wettbewerb und Herausforderung

Die Komplexität und Vernetztheit der verschiedenen Einstellmöglichkeiten ist riesig. Kleine Fehlentscheidungen können fatale Konsequenzen für die ganze Unternehmung haben. Diese Komplexität in den Griff zu bekommen ist sicherlich die grösste Herausforderung am ganzen Spiel.

5. Interaktionen

Damit nicht jeder Spieler mit dem „Zeigfinger in der Luft“ seine Einstellungen tätigt und das Unternehmen dadurch rote Zahlen schreibt, sind häufige Absprachen zwischen den einzelnen Teilnehmern unerlässlich. Diese Interaktionen können sehr vielfältig sein und benötigen einiges an Zeit (wenn man die ganze Aufgabe seriös anpacken möchte).

6. Repräsentation und Thema

Diverse Themen werden mit diesem CIO Business Game gemäss der offiziellen Webseite (www.cio-simulation.eu) angeschnitten:

- „Ganzheitliche Betrachtung eines Unternehmens mit hoher IT-Durchdringung.“
- „Unterschiedliche Entscheidungsdimensionen durch Projekte und Einzelkennzahlen.“
- „Förderung von Motivation, Identifikation und Begeisterung für den Themenbereich durch spielerische Auseinandersetzung. Teamarbeit ohne reales wirtschaftliches Risiko.“

Spielen und Lernen

Durch die Komplexität des Spiels wird die Zielgruppe klar definiert. Dies hat zur Folge, dass das vorliegende Rollen-/Simulations-Spiel nicht für jedermann geeignet ist. Da das CIO Business Game klar auf (zukünftige) Manager ausgerichtet wurde, ist es eine gute Trainingsplattform um das bisher Gelernte schadlos auszuprobieren und anzuwenden.

Es können folgende Lerninhalte ausgemacht werden:

- Prozesse, Prozeduren
In diesem Spiel werden alltägliche Prozesse verwendet wie Mitarbeiter einstellen, Preise erhöhen oder senken. Damit die Spieler den höchsten Eintrag in der High-Score-Liste erzielen können, ist es notwendig, dass alle Prozesse genau durchleuchtet und Abhängigkeiten festgestellt werden, damit der Kapital- und Mitarbeiterinsatz optimiert wird und der Gewinn der Autobank noch weiter steigt.
- Urteilsvermögen
Das Urteilsvermögen wird trainiert in dem man versucht abzuschätzen, welche Faktoren für die nächste Spielrunde relevant sein könnten und welche Parameter man aufgrund der vorhergehenden, verpatzten Runde anpassen muss.
- Systeme, Theorien
Nur wenn die Theorie, das Zusammenspiel der einzelnen Parameter, verstanden wird, können die Prozesse überhaupt analysiert werden. Diese theoretischen Grundlagen werden während des Spiels vermittelt: Dieses Spiel basiert auf den interaktiven Lernarten ‚Learning by Doing‘ und ‚Feedback Coaching‘. Eine scharfe Abgrenzung zwischen diesen und den oben aufgeführten Lerninhalten ist nahezu unmöglich.
- Kommunikation
Auch in diesem Mehrspieler-Rollenspiel ist die Kommunikation ein zentraler Faktor, welcher relevant dafür ist, ob die vier Spieler Erfolg oder Verlust ausweisen müssen.

2.4.3 Detektivspiele

Detektiv- (oder Krimi-) Spiele sind Gesellschaftsspiele mit mehreren Teilnehmern. Jeder Teilnehmer hat eine eigene Rolle (Ermittler, Kollege des Opfers, etc., und Mörder). Das Spiel startet mit der Aussage, dass ein Mord verübt wurde. In jeder Spielrunde erhält jeder Teilnehmer einen Zettel mit einem Hinweis über seine Person. In dem die Spieler miteinander diskutieren, werden in der Unterhaltung ungezwungen die Hinweise der anderen Mitspieler aufgedeckt. Jeder Mitspieler interpretiert die Situation mit Hilfe der Aussagen der anderen Teilnehmer und deren Hinweisen. Daraufhin stellt er passende Fragen um zusätzliche Informationen zu erhalten. Schliesslich gibt es eine Lösungsrunde. Jeder Teilnehmer erklärt nun, wer nach seiner Theorie das Verbrechen wie verübt hat und warum. Zum Ende der Lösungsrunde wird dann die Lösung aufgedeckt. In diesem Spiel gibt es nur eine Regel: Man darf nicht lügen (Host-Party.com 2006). Es gibt viele Abwandlungen dieses Spiels. Zwei davon habe ich nachfolgend aufgeführt.

Mystery at the Museum

Dieses Spiel wurde durch Wissenschaftler des MITs¹⁰ für das Wissenschaftsmuseum in Boston entwickelt. Die Hauptziele waren:

- Die Museumsbesucher tiefer in die Ausstellungen miteinbeziehen: Die Besucher sollen die Gegenstände erforschen und spezifischer darüber nachdenken, was sie zuvor nicht taten.
- Die Besucher sollen zwischen den einzelnen Gegenständen Verbindungen erkennen und Sachen entdecken, welche sie vorher nicht sahen.
- Am Ende sollen auch die gemeinsamen Diskussionen der Besucher angeregt werden. Der Ideen-Austausch untereinander soll verstärkt werden.

¹⁰ Massachusetts Institute of Technology - <http://www.mit.edu/> (09. August 2007)

Das mobile Detektivspiel ist wie folgt eingebettet: Die verrufene Bande der Flamingo-Diebe haben wieder zugeschlagen und ein wertvolles, unbezahlbares Objekt des Wissenschaftsmuseums gestohlen. Ein Team, bestehend aus einem Biologen, Wissenschaftler und Detektiv, hat nun die Aufgabe herauszufinden was die Diebe stahlen, wie sie es taten und schliesslich die Diebe zu erwischen bevor sie abhauen können.

Die Spieler haben viele unterschiedliche Möglichkeiten um Hinweise zu finden. Sie können virtuelle Figuren interviewen, Anhaltspunkte in den Ausstellungshallen sammeln, Proben mit virtuellen Instrumenten analysieren sowie die Informationen der Museumsexponate verwenden um ihre Aufgaben zu erfüllen. Das Spiel ist zu Ende, wenn die Spieler genügend Beweise gesammelt haben um die Schuldigen zu verhaften.

Die Spieler erhalten die ortsabhängigen Informationen über einen PDA und können untereinander und mit den Museumsexponaten via Infrarotübertragung Informationen austauschen. (Klopfer, et al. 2005)

1. Regeln

Alle Regeln sind in der Spielsoftware schon implementiert. Das Programm lässt also keine Aktionen zu, welche nicht erlaubt sind. Meine Vermutung ist, dass es eine Regel gibt, die besagt, dass nicht mit den anderen Teams gesprochen werden darf, da der Lerneffekt erhalten bleiben soll.

2. Ziel

Das Hauptziel ist, dass die Diebe gefangen werden bevor sie weg sind. Im Spiel wird das so umgesetzt, dass die Aufgaben innerhalb einer gewissen Zeit gelöst werden müssen.

3. Ergebnis und Feedback

Feedback wird durch die Kommunikation zwischen dem Spieler und dem PDA hergestellt. Das Gerät zeigt an, welche Proben man bereits gesammelt hat. Diese Informationen können an die anderen Mitspieler versendet werden. Es findet so ein Wissensaustausch statt. Das Gerät kann als „Wissensbehälter“ für alle gesammelten Informationen bezeichnet werden. Das Ergebnis ist die Antwort auf die Frage, ob die Diebe gestellt und der entwendete Gegenstand gesichert werden konnte oder nicht.

4. Wettbewerb und Herausforderung

Der Kampf gegen die Zeit ist nur ein kleiner Teil des Spiels. Die grösste Herausforderung ist, die gefundenen Gegenstände und Informationen so zu interpretieren, dass ein in sich schlüssiges Bild entsteht und man die Diebesbande überführen kann.

5. Interaktionen

Dadurch, dass jeder der drei Teilnehmer eine andere Funktion (Biologe, Wissenschaftler und Detektiv) inne hat, werden sie „gezwungen“ miteinander die gesammelten Informationen auszutauschen, neue Erkenntnisse zu ziehen, weitere zu erledigende Aufgaben bestimmen, etc.

6. Repräsentation und Thema

Das Thema dieses Spiels ist es, die Kombinationsgabe der Museumsbesucher und den Umgang mit neuen Technologien zu trainieren sowie sich mit den Inhalten des Wissenschaftsmuseums zu befassen.

Spielen und Lernen

Mystery at the Museum ist inhaltlich ein klassisches Detektivspiel. Darin werden folgende Lerninhalte umgesetzt:

- Fakten
Die Ausstellungsobjekte im Museum werden durch das Detektivspiel von den Spielern genauer unter die Lupe genommen als dies ausserhalb des Spiels der Fall wäre. Dadurch lernen die Spieler die Eigenschaften der Geräte spielerisch kennen.
- Fähigkeiten, Beobachten, Feststellen
Durch das Analysieren und Interpretieren von Hinweisen wird die Fähigkeit gefördert, ganzheitlich zu denken, kleinste Merkmale zu beachten und so zu kombinieren, dass die Diebe gefangen werden können.
- Urteilsvermögen
Aufgrund der gesammelten Hinweise ist ein Urteil zu fällen, wer nun die Diebe sind. Das Feedback am Ende des Spiels zeigt den Spielern auf, ob sie richtig lagen oder die Räuber schon längst über alle Berge sind.
- Kommunikation
Die vielen Interpretationen und Meinungen der einzelnen Spieler sind auf einen Nenner zu bringen, damit mit grösster Wahrscheinlichkeit die Beweise richtig interpretiert werden.

Charles River City

Auch dieses mobile Spiel wurde am MIT entwickelt. Das Basisszenario für dieses Spiel ist, dass mit einem Grossereignis in der Metro von Boston eine Krankheit ausbricht. Jedes Mal wenn das Spiel gespielt wird, modifiziert das System das Basisszenario leicht. Ein möglicher Ablauf ist folgender: 26 Fälle einer mysteriösen Krankheit sind aufgetaucht. Die Krankheit fegte durch Boston. Da in wenigen Wochen ein wichtiger, nationaler Politikanlass in die Stadt kommt, sind alle von diesem Problem betroffen (Bürger, Politiker, Gesundheitsbeamte). Die folgenden Fragen sind zu lösen: Was ist die Quelle der Krankheit? Ist dies ein Fall von einem biologischen Anschlag oder ein natürliches Ereignis? Wie beseitigt man am diese Krankheit am besten? Etc.

Die Spieler werden nun in Teams von 20 Experten aufgeteilt um dieses Problem zu lösen. Das Team beinhaltet Physiker, Gesundheitsexperten, Epidemiologen, Laboranten, Biologen, Informatiker und Umweltspezialisten. Diese Gruppe muss zusammenarbeiten um zuerst vorhandene Berichte und Überwachungsdaten auszuwerten. Anschliessend gilt es möglichst schnell die Quelle der Krankheit zu orten, einen grösseren Ausbruch zu verhindern und mit den privaten und öffentlichen Gemeinschaften zu kommunizieren um so effektive Interventionsmöglichkeiten zu schaffen. Das Team verdichtet die vorhandenen Informationen durch sammeln und analysieren von Proben, vorhandenen Patientengeschichten, etc. Das Team muss seine Erkenntnisse schnell gewinnen und zügig durchzuführende Tätigkeiten vorschlagen, um die Gefahr einzudämmen und das Risiko für die Personen und Umwelt so rasch als möglich zu verkleinern.

Das Spiel funktioniert mit einem PDA und GPS-basierten Ortungsinformationen. Zusätzlich gibt es noch sogenannte Non-Player-Character¹¹. Diese sind fest im Spiel eingebettet und können von Teammitgliedern via PDA befragt werden. Diese NPC können Polizist, Arzt oder einen anderen Beruf ausüben und auch ihre Aussagen und Positionen im Spiel ändern, so dass sie mal da, mal dort anzutreffen sind. Zusätz-

¹¹ NPC; virtuelle Figuren, welche von den Spielern befragt werden können.

lich kann ein Ereignis kaskadieren: Ein Ereignis löst andere Events aus. Es sind viele erdenkliche Möglichkeiten und Fälle im System eingebettet (Cheung 2003).

Analyse von Charles River City anhand der sechs Spielfaktoren

1. Regeln

Die Spielregeln sind alle in der Software implementiert. Das Programm lässt also keine Aktionen zu, welche nicht erlaubt sind. Meine Vermutung ist, dass es eine Regel gibt, die besagt, dass nicht mit den anderen Teams gesprochen werden darf, um den Lerneffekt aufrecht zu erhalten.

2. Ziel

Das Ziel des Spiels ist es die Krankheit zu bekämpfen, deren Ursache zu finden und den Ausbruch einzudämmen sowie Massnahmen für die Sicherheit der Bevölkerung zu ergreifen.

3. Ergebnis und Feedback

Die Spieler erhalten Informationen über die Anzahl erkrankten Personen und wie hoch die Verbreitung der Krankheit momentan ist. Das Ergebnis dieses Spiels wird sein, ob die Krankheit besiegt werden konnte oder ob die Stadt ausgestorben ist.

4. Wettbewerb und Herausforderung

Das Spiel ist ein Wettlauf gegen die Zeit. Ist das Team zu langsam ist die Stadt verloren und der Grossteil der Bewohner scheidet dahin. Damit dies nicht geschieht, müssen alle einzelnen Spezialisten miteinander zusammenarbeiten und so das Maximum aus den vorhandenen Informationen rausholen. Dies ist schon rein auf der Kommunikationsebene bei der gegebenen Teamgrösse eine grosse Herausforderung.

5. Interaktionen

Wie bei jedem Multiplayer-Game finden auch hier soziale Interaktionen zwischen den einzelnen Teilnehmern statt. Sicherlich wird auch nach dem Ende des Spiels noch über das Erlebte diskutiert.

6. Repräsentation und Thema

Das Thema dieses Spiels ist die Sensibilisierung der Spieler bezüglich des Umgangs mit Umwelt-ereignissen. Ein weiteres Thema ist das Training wie im Schadenfall die verschiedenen Spezialisten zusammenarbeiten sollen.

Spielen und Lernen

In diesem Simulationsspiel kann man erkennen, dass der Wettbewerbsgedanke fehlt. Es gibt keine Teams, welche gegeneinander kämpfen. Dafür steht der Gedanken des kollaborativen Arbeitens im Vordergrund. Es sind folgende Lerninhalte auszumachen:

- Theorien, Prozesse, Prozeduren

Die Haupttheorie, welche in diesem Spiel vermittelt wird, ist die Vorgehensweise in einem Katastrophenfall. Dieser Ablauf wird so vermittelt, dass die Spieler virtuelle Figuren befragen und logische Zusammenhänge suchen müssen.

- Fakten

Es können Fakten wie der Krankheitsverlauf eines Infizierten und Umweltschäden vermittelt werden. Dies ist sicher für Mediziner und Biologen eine interessante Methode ihren Lernstoff zu üben.

- Fähigkeiten

Die Spieler schulen ihre Fähigkeiten effiziente Interventionsmöglichkeiten zu ergreifen und innert kurzer Zeit adäquat zu reagieren.

- Kommunikation

Nur wenn die Teilnehmer zusammenarbeiten ist es möglich die Opferzahl tief zu halten. Diese Zusammenarbeit ist eine Herausforderung, müssen doch rund 20 Experten miteinander kommunizieren. Es ist eine geeignete Führungsstruktur aufzubauen, um die vorhandene, knappe Zeit nicht zu zerreden sondern aktiv zu nützen.

2.4.4 „Multiple-Choice-Spiele“

Multiple-Choice-Spiele stellen dem Spieler Fragen und belohnen richtige und bestrafen falsche Antworten. Das nachfolgende Spiel erläutert eine mögliche Variante.

Musex

Musex wurde für Kinder entwickelt um im Museum die Ausstellung durch gemeinsames Entdecken kennenzulernen. Die Kinder interagieren dadurch mit den ausgestellten Objekten viel aktiver.

In der Mitte des PDAs ist ein Bild, welches in zwölf einzelne weisse Felder eingeteilt und mit Nummern verdeckt ist. Ziel ist es, dass das Foto komplett angezeigt wird. Hinter jeder Nummer befindet sich eine Multiple-Choice-Frage mit einem Antwortstamm von vier Alternativen. Die Fragen beziehen sich jeweils auf die Ausstellungsobjekte. Wird die Frage korrekt gelöst, verschwindet das weisse Feld und ein Teil des Bildes erscheint. Je mehr Fragen korrekt beantwortet werden, desto vollständiger wird das Bild aufgedeckt.

Zu Beginn des Spiels werden Zweier-Paare gebildet und die beiden PDAs miteinander verlinkt. Jeder Spieler sieht nun auch die Antworten des jeweils anderen. Hat beispielsweise ein Kind die Frage richtig beantwortet, wird auf beiden Bildschirmen das weisse Feld entfernt und ein Teil des Bildes kommt zum Vorschein. Löste ein Spieler die Frage falsch, wird auf beiden PDAs das Feld grau abgedeckt. Um miteinander zu interagieren und diskutieren, erhält jeder Spieler zusätzlich eine Funkgarnitur (Yatani 2004).

Analyse von Musex anhand der sechs Spielfaktoren

1. Regeln

Auch hier sind Regeln ins System eingebettet. So wird bei falschen Antworten nicht das Bild aufsondern grau abgedeckt.

2. Ziel

Ziel dieses Spiels ist das Bild komplett aufzudecken. Das Ziel wird durch das Kind erreicht oder nicht erreicht.

3. Ergebnis und Feedback

Konnte eine Multiple-Choice-Aufgabe nicht korrekt gelöst werden, zeigt das System sofort die richtige Lösung und die dazugehörige Erklärung an. Durch die Interaktionen mit dem Gerät wird erzielt, dass das Kind auf eine spielerische Art und Weise einerseits mit der Bedienung des PDAs vertraut wird und andererseits sich mit der Ausstellung beschäftigt, was ja der Hauptgrund des Museumsbesuchs ist.

4. Wettbewerb und Herausforderung

Die Herausforderungen sind die nicht immer ganz einfachen Fragen sowie auch die japanischen Schriftzeichen, welche von den Kindern noch nicht immer verstanden werden (Yatani 2004).

5. Interaktionen

Da ein Spieler immer von einem weiteren Spieler abhängig ist und umgekehrt, werden von der Software Interaktionen zwischen den Kindern schon beinahe erzwungen. Yatani konnte aber bei seinen Untersuchungen beobachten, dass sich die Kinder häufig die Fragen aufteilten, aber durch die synchronen Bildabgleiche immer wussten, was der andere Spieler im Moment tat.

6. Repräsentation und Thema

Bei diesem Spiel handelt es sich um das Thema „Lernen und Entdecken im Museum“. Die Fragen werden so gestellt, dass die Kinder die von den Lehrpersonen definierten Lernziele erreicht.

Spiele und Lernen

Dieses Spiel ist ein Lernspiel, das speziell für sehr junge Schüler entwickelt wurde. Daher ist auch der Komplexitätsgrad relativ tief. Wie eingangs in der Theorie zu den sechs Faktoren erläutert, ist es wichtig, dass das Spiel den Spielern angepasst ist, damit die Spieler nicht die Freude daran verlieren. Dies wird hier mit der vorliegenden Aufgabenstellung erreicht. Folgende Lerninhalte lernen die Schüler mit diesem Spiel:

- Fakten

Da die Fragen spezifisch zu einzelnen Ausstellungsobjekten gestellt werden, lernt es die (wichtigsten) Fakten zu diesen Exponaten auf eine spielerische Art und Weise kennen.

- Kommunikation

Da die Schüler noch so jung sind, lernen sie schon sehr früh im Team zu arbeiten. Dies ist förderlich für ihre spätere Zukunft in dem sie den Umgang mit anderen schon kennen und dadurch nicht egoistisch handeln, aber trotzdem wissen wie sie sich durchsetzen können.

2.5 Lernprozess

Die folgenden Ausführungen sind angelehnt an die Ausbildungsmethodik der Schweizer Armee. Damit der militärische Unterton nicht überhand nimmt, habe ich einzelne Begriffe in andere sinngemässe Ausdrücke umbenannt.

Ich betrachte nun den Lernprozess einer Übung wie sie die Lehrperson gestalten könnte. Dieser Lernprozess ist aufgrund der erhaltenen Aufgabenstellung elementar für die komplette nachfolgende Arbeit. Ein Lernprozess ist vielfach in die folgenden Elemente unterteilt:



Abbildung 2: Der Lernprozess (eigene Darstellung)

- Aufgabenstellung

Der Lehrer erstellt die Aufgaben, welche der Schüler in einer Übung oder in einer Klausur zu lösen hat. Sind die Aufgaben erstellt, übergibt sie der Lehrer mündlich oder schriftlich dem Schüler. Etwas weiter unten werden die verschiedenen Arten der schriftlichen Aufgaben erörtert.

- Lösen der Aufgabe

Der Schüler löst die Aufgabe in dem er sein Wissen abrufen, in Lexika nachliest, er sich kreativ betätigt oder etwas gestaltet und dichtet und so versucht, die Aufgabe möglichst genau und gut zu lösen. Ist er der Meinung, die Aufgabe sei nun gelöst oder ist die vom Lehrer vorgegebene Zeit verstrichen, gibt er seine Lösung dem Lehrer ab.

- Korrektur / Bewertung

Die Lehrperson hat vom Schüler die Lösungen erhalten und korrigiert diese nun. Dazu kann er sich einer vorher erstellten Musterlösung bedienen oder er vergleicht die Lösung anhand von anderen Faktoren wie der Leistungsfähigkeit des Schülers und kann so individuell die Leistung des Schülers aufgrund dessen Möglichkeiten beurteilen.

- Feedback

Der Lehrer gibt dem Schüler eine Rückmeldung über die Lösung: wie gut sie war, was er das nächste Mal besser machen kann sowie welche Teile er sehr gut gelöst hat.

2.5.1 Aufgabenstellungen

Ich werde in diesem Abschnitt auf die verschiedenen möglichen Aufgabenarten bei schriftlichen Fragestellungen eingehen. In der Aufgabenstellung der vorliegenden Diplomarbeit geht es in erster Linie darum, Fragestellungen subjektiver Art in den mExplorer zu integrieren. Damit aber der Leser ein nahezu vollständiges Bild über die im mExplorer verwendeten Aufgabenarten erhält, werden auch die anderen Fragearten kurz beleuchtet.



Abbildung 3: Übersicht „Schriftliche Fragestellungen“ (eigene Darstellung)

Bei den schriftlichen Fragestellungen werden zwei Arten unterschieden: Einerseits Fragen, welche maschinell ausgewertet werden können (objektiv) und andererseits Aufgaben, deren Lösungen ein Computer nicht korrigieren kann sondern der Lehrer von Hand mit seinem Fachwissen (subjektiv) beurteilen muss. Es sind bewusst nicht alle möglichen maschinell auswertbaren Fragestellungen aufgeführt, da beinahe alle „Unterarten“ wie beispielsweise Reihenfolge- oder Zuordnungsaufgaben auf dieselbe Art vom Rechner korrigiert werden wie Short-Answer- und Multiple-Choice-Aufgaben.

2.5.1.1 Objektive Fragestellungen

Objektive oder maschinell-auswertbare Fragestellungen können deshalb von einem Rechner korrigiert werden, weil die Lösung „einfach“ ist bzw. der Computer nur zwischen richtig und falsch unterscheiden muss. Er muss keine Semantik¹² hinter den einzelnen Zeichen verstehen, sondern er macht einen trivialen Vergleich ob die Eingabe des Schülers mit den Werten der Musterlösung übereinstimmt (Jacobs, Übungsaufgaben stellen mit JavaScript 2005).

Maschinell-auswertbare-Aufgaben werden im Alltag aufgrund der einfachen Korrektur sehr oft eingesetzt. Praktisch alle Lehrziele lassen sich damit erfassen. Auch die Erreichung von Verständnis- und Anwendungs-Lehrziele können damit geprüft werden. Einzig kreative Leistungen lassen sich mit diesen Aufgabentypen nicht erfassen. (Jacobs, Guidelines zur Konstruktion von Multiple-Aufgaben-Aufgaben 2005).

¹² Semantik: Sinn, Bedeutung

Wo nichts anderes erwähnt ist, werden die Erläuterungen der beiden nachfolgenden Aufgabentypen an Gronlund angelehnt.

Multiple-Choice-Aufgaben

Multiple-Choice-Aufgaben werden sehr häufig eingesetzt. In psychologischen Tests (z.B. Intelligenztests), bei theoretischen Auto-Fahrprüfungen, bei Klausuren an der Universität, etc. praktisch über all wo Wissen überprüft wird, trifft man auf solche Aufgaben. Nachfolgend ein Beispiel zur Veranschaulichung dieses Aufgabentyps:

Wie heisst die Hauptstadt von Frankreich? (*Aufgabenstamm*)

- ☐ 1. Madrid (*Distraktor*)
- ☐ 2. Bern (*Distraktor*)
- ☐ 3. Paris (*korrekte Antwort*)
- ☐ 4. Amsterdam (*Distraktor*)

Tabelle 2: Beispielfrage für eine Multiple-Choice-Aufgabe

Bei der Erstellung des Aufgabenstamms kann zwischen zwei Varianten ausgewählt werden:

1. Der Aufgabenstamm wird als Frage formuliert. Die Alternativen sind jeweils die Antworten auf die Frage (siehe oben).
2. Der Aufgabenstamm ist ein unvollständiger Satz, der durch jede einzelne Alternative vervollständigt wird („Die Hauptstadt von Frankreich heisst...“).

Bei der Erstellung der Antwortalternativen kann wiederum zwischen zwei Varianten unterschieden werden:

1. Die korrekte Antwort ist wahr. Alle anderen Alternativen (Distraktoren) sind falsch.
2. Die korrekte Antwort ist die beste Alternative unter allen Auswahlmöglichkeiten.

Bei der Erstellung der Multiple-Choice-Aufgabe ist die Wahl von plausiblen Distraktoren das grösste Problem. Denn je besser die Distraktoren, desto höher ist die Wahrscheinlichkeit, dass ein unwissender Schüler eine falsche Alternative wählt und so eine faire Bewertung erhält. Als Ratewahrscheinlichkeit bezeichnet man die Wahrscheinlichkeit, dass der Schüler die richtige Antwort errät. Je kleiner diese Wahrscheinlichkeit ist, desto höher ist die Qualität des Tests. Die Ratewahrscheinlichkeit ist abhängig von der Anzahl Alternativen und der Plausibilität der Distraktoren. Da der Schüler kein Zufallsgenerator ist, kann er nicht plausible Alternativen sofort ausschliessen und erhöht somit die Ratewahrscheinlichkeit. Dies gilt es zu verhindern. Als mögliche Hinweise für den Schüler werden von Gronlund folgende aufgezählt. Diese Hilfestellungen sind nach Möglichkeit zu umgehen:

- Ist die korrekte Antwort nahe zu aus dem Lehrbuch kopiert, sind diese dem Schüler irgendwie bekannt. Dass der Schüler aber die Definition verstanden hat, ist nicht zwingend der Fall.
- Die korrekte Antwort sollte nicht ausführlicher sein als die Distraktoren. Falsche und richtige Antworten sollen sich nicht nur anhand des Erscheinungsbildes unterscheiden lassen.
- Es gibt ähnliche Wörter im Aufgabenstamm und in der korrekten Antwort.
- Absolute Begriffe wie „kein, ausschliesslich, immer, niemals“ implizieren häufig die Falschheit der Aussage. Dagegen verweisen Begriffe wie „manchmal, häufig, gewöhnlich“ tendenziell auf die richtige Antwort.
- In einer Multiple-Choice-Frage, bei der es nur eine richtige Antwort gibt, sorgen Beziehungen zwischen zwei Antwortmöglichkeiten dafür, dass die Anzahl echter Distraktoren verkleinert wird.

Haben zwei Distraktoren nahezu dieselbe Bedeutung, müssen beide falsch sein, da nur eine Option korrekt ist.

Die Lehrperson erzielt den grössten Lerneffekt, wenn in den Distraktoren klassische Fehler der Lernenden eingebaut sind. Je grösser der Erfahrungsschatz des Lehrers, desto mehr kennt er die Fehler, welche die Schüler machen, desto besser kann er „naheliegende“, aber falsche Antworten in Distraktoren einflechten.

Wie korrigiert nun ein Computer solche Multiple-Choice-Aufgaben? Die Auswertung durch den Rechner kann folgendermassen programmiert werden: Der Schüler klickt die zweite Alternative an, die er für richtig hält, und drückt auf die Schaltfläche „Antwort absenden“ (nicht dargestellt). Der Computer erhält die Ziffer „2“ zum Vergleichen. (Anmerkung: jede Antwortalternative ist jeweils mit einer einmaligen Zahl verknüpft). Er überprüft nun, ob die Schülerantwort der Musterlösung entspricht ($2 == 3 ?$ [$==$: entspricht]). Dies ist nicht der Fall, daher gibt der Rechner in diesem Beispiel „falsch“ zurück. Hätte der Schüler aber die Antwort „3“ (Paris) ausgewählt, wäre der Vergleich ($3 == 3 ?$) positiv ausgefallen und der Rechner würde „richtig“ zurückliefern.

Short-Answer-Aufgaben

Short-Answer- oder Lückentext-Aufgaben gehören zu den Arten der offenen Beantwortungsaufgaben und sind in dieser Kategorie der restriktivste Typ. Es wird zwar eine freie Beantwortung ermöglicht, jedoch sind nur sehr kurze Antworten erlaubt, welche praktisch nie länger als ein Wort oder eine Zahl sind. Der Vorteil von Short-Answer-Aufgaben im Vergleich zu den Multiple-Choice-Aufgaben ist, dass die Ratewahrscheinlichkeit entfällt und so die Aufgabenschwierigkeit erhöht wird, da nur gelerntes Wissen abgefragt wird. Short-Answer-Aufgaben können verwendet werden um Faktenwissen, Verständnis und Verstehen sowie die Anwendung des Lernstoffes (meistens Berechnungen) zu überprüfen (Jacobs, Guidelines zur Erstellen von Short-Answer-Aufgaben 2003). Ein Beispiel zur Veranschaulichung:

Die Hauptstadt von Frankreich heisst?

Tabelle 3: Beispielfrage für eine Short-Answer-Aufgabe

Auch zu Lückentextaufgaben führt Gronlund Tipps und Ratschläge auf:

- Ein Satz - eine Lücke: Es sollte höchstens eine Lücke in einem Satz geben. Mehrfachlücken in derselben Aufgabenstellung sind zu vermeiden, da so unerwünschte Abhängigkeiten entstehen können. Wenn die erste Lücke nicht beantwortet werden kann und dieses Wissen aber notwendig ist für das korrekte Verständnis der zweiten Aufgabe, verunmöglicht dies dem Schüler die zweite Lücke zu füllen, obwohl eigentlich das Wissen vorhanden wäre.
- Wird die Lücke ans Ende des Satzes platziert, wird die Aufgabe für den Nutzer einerseits klar strukturiert und andererseits lässt sich die korrekte Antwort am optimalsten finden.
- Die Frage soll so formuliert werden, dass möglichst nur eine einzelne (kurze) Antwort zutreffend ist. Dies ist vielfach die grösste Schwierigkeit bei Lückentexten, da nicht selten mehr als eine zutreffende Antwort möglich ist. Durch die Verwendung bestimmter Artikel wie Plural, Singular, Geschlecht können plausible Alternativen einfach in der Aufgabenstellung ausgeschlossen werden.

Ansonsten sind die Short-Answer-Aufgaben den Multiple-Choice-Fragen sehr ähnlich. Dadurch haben die oben erwähnten Hinweise und Ratschläge auch bei Lückentextaufgaben ihre Gültigkeit.

Wenn diese Frage von Tabelle 3 auf dem Schülercomputer angezeigt wird, gibt ein wissender Schüler „Paris“ ein. Der Computer wiederum nimmt die Eingabe und vergleicht sie mit der Musterlösung: („Paris“ == „Paris“ ?). Er vergleicht ob die Buchstabenreihenfolge der Schülereingabe mit der Musterlösung übereinstimmen. Die Ausgabe des Computers wird „richtig“ sein.

Die eine Hauptschwierigkeit bei der maschinellen Auswertung dieses Aufgabentyps ist, dass je nach Fragestellung, es korrekte Synonyme gibt und der Computer auch diese als richtig zu anerkennen hat. Diese Synonyme müssen alle vorher vom Lehrer eingegeben werden. Um so wenig verschiedene Antworten wie möglich zu zulassen, ist es notwendig, dass der Lehrer die Frage möglichst genau auf den gesuchten Begriff einengt ohne aber zu viele Hinweise dem Schüler zu geben.

Ein weiteres Problem ist die Rechtschreibung: nicht nur soll der Rechner „Paris“ als richtig anerkennen, sondern auch „PARIS“ und „paris“. Der Computer muss also „case insensitive“ korrigieren (unabhängig von der Gross- und Kleinschreibung). Ein weiteres Problem sind schreibschwache Schüler: Es wird immer Schüler geben, die einen Rechtschreibfehler machen und anstatt „Paris“ „Baris“ eingeben. Der Schüler weiss in diesem Fall also, wie die Hauptstadt heisst, das Lernziel ist erfüllt, aber der Computer bewertet diese Lösung trotzdem als falsch. Auch hier ist es möglich, dass der Lehrer verschiedene als richtig zu anerkennende Begriffe erfasst. Der Arbeitsaufwand alle diese Möglichkeiten in den Computer einzugeben ist bei vielen Short-Answer-Aufgaben aber tendenziell unverhältnismässig. Vor allem wenn die Fragen nur bei einem Test verwendet werden.

2.5.1.2 Manuell-auswertbare Fragestellungen

Manuell-auswertbare Fragestellungen sind Aufgabenstellungen, welche nicht von einem Computer korrigiert werden können sondern ausschliesslich von Menschen. Solche Fragestellungen bieten sehr viel Raum für die Beantwortung. Es gibt meistens mehrere richtige Varianten. Der Schüler antwortet in Sätzen. Der Computer kann die Lösungen solcher Fragestellungen nicht auswerten, weil er nicht die Semantik hinter den einzelnen Wörtern versteht. Während er bei den oben beschriebenen Aufgabenstellungen lediglich die Lösung vergleicht, ob die Buchstabenreihenfolge dieselbe ist wie in der Musterlösung, müsste er hier den Sinn der Wörter und noch weiteres verstehen und dann mit dem Sinn der Musterlösung vergleichen können. Soweit ist die Technik aber heute noch nicht. Wird bei solchen Fragestellungen trotzdem ein Rechner eingesetzt, dient er lediglich als „Übertragungsmittel“ zwischen Schüler und Lehrer. Er überträgt die Antwort vom Schüler- zum Lehrercomputer.

Kreativaufgaben

Essay-Test-Fragestellungen oder Kreativaufgaben gehören in die Kategorie der offenen Aufgaben und erlauben dem Schüler eine freie Beantwortung. Es ist angebracht sie zu verwenden, wenn der Lehrer das Finden und Darstellen kreativer Ideen überprüfen will, der Schüler die Lösung selber strukturieren und das gelernte Wissen integrieren und bewerten soll. Mit diesen Aufgaben wird vom Schüler ein tieferes Verständnis über den Wissensstoff und die Auseinandersetzung mit komplexen Problemen verlangt. Häufig sind solche Aufgaben ähnlich wie reale Problemstellungen gestaltet (Jacobs, Guidelines zur Konstruktion von Essay-Test-Aufgaben 2000).

Ein Beispiel einer solchen (restriktiven) Essay-Test-Aufgabe:

Vergleiche die IEEE-Protokolle 802.11 (WLAN) und 802.3 (Ethernet) miteinander anhand von drei selber gewählten Kriterien.

Tabelle 4: Beispielfrage für eine Essay-Test-Aufgabe

Es können zwei Varianten unterschieden werden (Gronlund 1998):

1. Restriktive Antwortform

Die Antworten des Studenten werden schon in der Aufgabenstellung eingeschränkt. (Beim vorliegenden Beispiel: „drei (...) Kriterien“.) Der grosse Vorteil ist, dass dadurch eine gewisse Objektivität bei der Auswertung erzielt werden kann, da für die zutreffenden korrekten Antworten bereits eine gewisse Bandbreite abgesteckt wurde.

2. Erweiterte Antwortform

Diese Aufgabenstellung ist für den Schüler sehr weit offen und er kann das Thema nach eigenen Vorstellungen ausarbeiten. Einschränkungen sind häufig formaler Natur und werden von der Lehrperson vielfach über die Zeit (maximaler Aufwand: zwei Stunden) oder den Umfang (maximal eine A4 Seite) definiert. Die grösste Schwierigkeit befindet sich in der Korrektur dieser Aufgaben: Die Lösung wirklich objektiv zu bewerten ist eine grosse Herausforderung.

Geeignet sind solche Aufgaben vor allem bei der Erfassung komplexerer Lernziele. Zusätzlich sind diese Aufgaben auch nützlich wenn mehr der Übungs- als der Klausuraspekt im Vordergrund steht. Eine objektive Beurteilung ist bei Übungen häufig nur von kleiner Relevanz: Der Trainingseffekt steht im Vordergrund.

Für einfachere Lernziele sind Multiple-Choice- und Short-Answer-Aufgaben vorteilhafter, da sie einfacher zu korrigieren sind und es trotzdem keine Einbussen bezüglich der Objektivität in der Korrektur nach sich zieht (Jacobs, Guidelines zur Konstruktion von Essay-Test-Aufgaben 2000).

2.5.2 Korrektur und Bewertung

Ab diesem Abschnitt beziehen sich alle nachfolgenden Aussagen und Feststellungen nur noch auf manuell-auswertbare Fragestellungen.

Der nächste Schritt, welcher auf das Lösen der Aufgabe durch den Schüler folgt, ist die Korrektur durch den Lehrer. Der Lehrer korrigiert die Aufgabe anhand einer Musterlösung, welche er für die Essay-Test-Aufgaben erstellt hat. Er entscheidet, ob die Schülerlösung richtig, teilweise richtig oder falsch ist. Anhand dieser Korrektur ist eine anschliessende Bewertung der Schülerleistung möglich.

Damit eine Leistung beurteilt werden kann, müssen gewisse Bezugsnormen zum Vergleichen vorhanden sein. Es werden grundsätzlich drei Bezugsnormen für die Bewertung unterschieden (Sacher 1996):

1. Sozialnorm

Die Leistung des Einzelnen wird Verglichen mit den Leistungen der anderen Lernenden in der Gruppe.

2. Curriculare oder kriteriale Norm

Der Lehrer beurteilt, inwieweit der Schüler die gesetzten Lernziele erreicht hat. Diese Bewertung erfolgt unabhängig von den anderen Schülern in der Gruppe. Sie wird nur mit dem Erreichen des gesamten Lernziels verglichen. Es kann also eine Prozentzahl berechnet werden, wie viele Prozentpunkte des maximal Möglichen der Schüler erreicht hat.

3. Individualnorm

Die Individualnorm vergleicht die Leistungssteigerung des einzelnen Schülers. Zuerst legt der Schüler den Test zum ersten Mal ab. Danach lernt er den Stoff nochmals und schliesslich absolviert er denselben Test zum zweiten Mal. Nun vergleicht der Lehrer die Anfangs- und Endleistung miteinander.

Die nächsten drei Unterkapitel behandeln mögliche Arten wie Kreativaufgaben bewertet werden können.

2.5.2.1 *Rating Score*

Die Punkteverteilung wird anhand vordefinierter Kriterien und der Musterlösung durchgeführt. Sollten beispielsweise fünf Begriffe erforderlich sein, um bei der Aufgabe die maximale Punktzahl zu erreichen. Werden aber nur drei richtige Begriffe vom Schüler aufgeführt, erhält er auch nur drei von fünf Punkte. Dieses Vorgehen wird bei jeder einzelnen Kreativaufgabe wiederholt. Schliesslich kann man die Anzahl Punkte aufaddieren und erhält so einen Wert, welcher nun mit den anderen Schülern oder der maximal möglichen Punktzahl verglichen werden kann.

2.5.2.2 *Nominalskala*

Eine weitere Möglichkeit Punkte zu verteilen ist, dies anhand einer Likert-Skala vorzunehmen in dem man die Schülerlösung mit Worten bewertet und so mit den Lösungen der anderen Schüler vergleichen kann.

Beispiel: Wir alle kennen die Bewertungsbezeichnungen von der eigenen Schulzeit her:

Hervorragend (6)
Gut (5)
Ausreichend (4)
Ungenügend (3)
Schwach (2)
Sehr schwach (1)

Tabelle 5: Notenübersicht aus (IfI 2001) und (ZFH 2005)

Diese Skala ist eine Nominalskala. Sie sagt nichts darüber aus, ob der Abstand zwischen zwei Werten überall gleichgross ist. Sie sagt nur aus, dass ‚hervorragend‘ besser ist als ‚gut‘ und ‚gut‘ besser als ‚ausreichend‘. Obwohl diese Tatsache bekannt ist, setzen sich die Lehrpersonen aus praktischen Gründen darüber hinweg und vierteln die Zwischenschritte (Lütgert 2004). Diese Skala wird verwendet, wenn mit der kriterialen Bezugsnorm beurteilt wird.

Auch ist es möglich mit den Bewertungen „ist besser als“, „ist gleich gut wie“ und „ist schlechter als“ eine Reihenfolge der Schülerlösungen herzustellen. Eine mögliche Aufgabe bei der diese Skala angewendet werden kann ist, wenn die Schüler ein vier-zeiliges Gedicht zum Thema X schreiben sollen. Hier kann der Lehrer keine Musterlösung erstellen, wird doch dem Schüler die Beantwortung dieser Aufgabe sehr stark frei gestellt und nur durch das Thema und die vier Zeilen eingeschränkt. Die Schwierigkeit dieser Skala ist, dass sie die Subjektivität des Korrektors nicht ausschaltet. Was ihm am besten gefällt erhält die höchste Punktzahl.

Am besten kann diese Skala eingesetzt werden, wenn ein Gremium (zum Beispiel alle Schüler der Klasse) über die Lösungen (Gedichte) entscheidet und so die verschiedenen Lösungen miteinander vergleicht. Die Subjektivität des Einzelnen kann so sehr stark reduziert werden.

2.5.2.3 *Scoring Rubric*

Die folgenden Ausführungen zu diesem Korrektur- und Bewertungsschema sind stark an Moskal angelehnt. Scoring Rubrics werden verwendet, wenn ein Urteil über die Qualität einer Lösung gefällt werden soll. Dazu wird ein deskriptives Bewertungsschema erstellt, welches benützt wird um die Analyse der Schülerlösung durchzuführen. Durch das vordefinierte Lösungsschema wird die Subjektivität im Bewertungsprozess verkleinert. Die Wahrscheinlichkeit steigt, dass zwei unabhängige Korrektoren die selbe Prüfung gleich bewerten.

Beispiel eines solchen Schemas für die Korrektur eines Aufsatzes:

Kriterium	Ungenügend 0 Punkte	Ausreichend 5 Punkte	Gut 10 Punkte
Inhalt und Entwicklung (70%)	Inhalt ist unvollständig, Hauptaussagen sind nicht klar oder nicht überzeugend.	Inhalt umfasst nicht alle Punkte und/oder ist nicht überzeugend. Hauptaussagen sind aufgeführt, aber zu wenig entwickelt. Inhalt ist inkonsistent im Hinblick auf die Zielsetzung.	Inhalt ist allumfassend, präzise und überzeugend. Hauptaussagen sind klar und gut mit Argumente unterstützt.
Organisation und Struktur (30%)	Organisation und Struktur gehen nicht mit den Aussagen des Autors einher. Einführung oder Konklusion fehlt.	Der Struktur des Textes ist nicht einfach zu folgen. Einführung fehlt oder gibt keine Vorschau auf wichtige Aussagen. Abschnittsübergänge sind zu verbessern. Konklusion folgt nicht dem Fluss des vorhergehenden Textes.	Der Struktur des Textes ist ersichtlich und einfach zu folgen. Einführung liefert genügend Hintergrundinformationen zum Thema und hält eine Vorschau auf wichtige Aussagen. Abschnitte sind vorhanden und sinnvoll untergliedert. Konklusion ist logisch und folgt dem Fluss des vorhergehenden Textes.

Tabelle 6: Beispiel Scoring Rubric.

In Anlehnung an http://upload.wikimedia.org/wikipedia/en/5/59/Graded_rubric.jpg (zuletzt geöffnet am 08. August 2007)

In dieser Scoring Rubric sind maximal 10 und minimal 0 Punkte zu erreichen.

Wie verwendet man nun solch eine Scoring Rubric? Der Lehrer liest den Aufsatz des Schülers durch und kreuzt anschliessend die zwei passenden Felder auf der Tabelle an. Die Punkte des Schülers berechnet er, in dem er pro Kategorie die Gewichtung mit den erreichten Punkten multipliziert und anschliessend alle so gewonnenen Resultate aufaddiert. In diesem Beispiel wäre wie folgt zu rechnen:

$$\begin{aligned}
 5 \text{ Pkt} \cdot 70\% &= 3.5 \text{ Pkt} \\
 10 \text{ Pkt} \cdot 30\% &= 3 \text{ Pkt} \\
 3.5 \text{ Pkt} + 3 \text{ Pkt} &= 6.5 \text{ Pkt}
 \end{aligned}$$

Dieser Schüler hat also von 10 möglichen 6.5 Punkte erzielt. Man neigt nun zu sagen, dass diese Methode einfach eine andere Art ist Punkte zu verteilen und im Prinzip einer Musterlösung gleichkommt. Dies ist aber nicht der Fall: Der Hauptnutzen liegt beim Feedback an den Schüler. Der Lernende erhält mit dieser Methode klar angezeigt, was er hätte besser machen können. Wird bei einem normalen Bewertungsverfahren dem Schüler einfach mitgeteilt, dass er 80 von 100 Punkten erzielte ohne zu sagen was er falsch machte, ist bei der Übergabe dieses Scoring-Rubric-Formulars für den Schüler klar ersichtlich was er mehr hätte leisten müssen um die maximale Punktzahl zu erreichen. Scoring Rubrics geben für jedes Bewertungskriterium bekannt was gefordert wird. Der Lernende zieht aus der Bewertung seine Schlüsse und kann beim nächsten Aufsatz seine Leistung verbessern.

Scoring Rubrics können für schriftliche und mündliche Aufgaben verwendet werden. So wird dem Schüler beispielsweise bei einer Präsentation klar mitgeteilt, wo seine Stärken und Schwächen liegen. Auch können erstellte Scoring Rubrics immer wieder verwendet werden. Der Schüler kann so über all seine Vorträge seine Leistungssteigerung nachvollziehen und sieht, was für Anforderungen die nächst höhere Stufe stellt.

Bei der Erstellung einer Scoring-Rubric müssen in einem ersten Schritt klar die Qualitäten identifiziert werden, welche die Schülerlösung beinhalten muss. Anhand dieser Qualitäten werden dann die Kriterien ausgewählt. Die Kriterien müssen leicht voneinander abzugrenzen und sollten zusätzlich noch messbar sein. Es sollte bei der Korrektur also einfach sein die einzelnen Kriterien auseinander zu halten. Die Anzahl der unterschiedlichen Qualitätsstufen ist nicht vorgegeben und kann von der Lehrperson selber bestimmt werden (in diesem Beispiel: drei Qualitätsstufen: ungenügend, ausreichend, gut).

Es ist gut vorstellbar, dass diese Scoring Rubric nur digital verwendet wird. Eine mögliche Idee von mir ist folgende: Für die Erstellung der Kriterien mit den gewünschten Rubriken wird ein spezielles Programm verwendet. Bei der Korrektur kann der Lehrer nur noch den Namen des Schülers und dessen Mailadresse eingeben und anschliessend die entsprechenden Felder der Tabelle anwählen. Mit einem Klick auf „Speichern/Senden“ würde dann das Resultat einerseits auf dem Computer des Lehrers gespeichert und andererseits direkt dem Schüler via Email versandt.

2.5.3 Feedback

Als letzte Phase einer Klausur oder Übung folgt das Feedback vom Lehrer an den Schüler. Es geht darum, dem Schüler die erreichte Note und erhaltenen Punkte mitzuteilen und andererseits ihm zu kommunizieren, was er gut löste, wo noch Verbesserungspotential vorhanden ist und wie die richtige Antwort gelautet hätte.

Wie mir Dirk Frohberg im Interview mitteilte, ist die Feedback-Phase optional. Es ist möglich, dass der Lernende eine Aufgabe erhält (zum Beispiel: „Bastle eine Handtasche für deine Mutter zum Muttertag.“), diese dann löst und die Lehrperson diese Aufgabe aber nicht bewertet. In diesem Fall ist auch kein Feedback durch den Lehrer notwendig.

Feedback ist die Rückmeldung in dem bestimmte Personen (hier: Lehrperson) im Gespräch oder schriftlich über die Leistung einer anderen Person urteilen und diese bewerten. Häufig lernt der Feedback-Nehmer (Schüler) neue Sichtweisen kennen und erhält zusätzliche Informationen. Der Schüler kann damit sein Verhalten ändern, seine Fähigkeiten erweitern sowie Lehren aus allfälligem Fehlverhalten ziehen (Menzel 2005).

Es werden zwei Formen von Feedback unterschieden: unmittelbares und verzögertes Feedback.

1. Unmittelbares Feedback

Unmittelbar nach dem Beantworten der Aufgabe wird dem Schüler die richtige Lösung angezeigt bzw. dargelegt, welche Fehler er beging.

2. Verzögertes Feedback

Dem Lernenden wird erst nach einem gewissen zeitlichen Abstand Feedback gegeben. Dieser zeitliche Abstand kann wenige Minuten aber auch einige Stunden sein.

Anderson und Kulhavy zeigen mit ihrer perseveration-interference-Theorie auf, dass verzögertes Feedback in bestimmten Kontexten dem unmittelbaren überlegen ist. Die Theorie erklärt, dass bei unmittelbarem Feedback der Lernende in dem Zustand verbleibt in dem er die falsche Antwort gegeben hat. Die Erinne-

rung an die falsche Antwort ist stärker als die Information über die korrekte Lösung. Bei verzögertem Feedback ist diese Gefahr nicht vorhanden, da der Schüler die abgegebene falsche Antwort zwischenzeitlich vergessen hat. So fällt es dem Schüler leichter die richtige Lösung zu behalten.

Es gibt verschiedene Arten von Lehrerkomentaren (Feedbacks) zu Leistungen. Über diese schreibt Krampen folgendes:

1. Sozial-orientiertes Feedback
(Lehrer ruft vor der Klasse aufgrund der schlechten Klassenleistung aus.)
Dieses Feedback wirkt bei schwächeren Schülern deutlich negativ während sich bei leistungsstärkeren dieser Lehrerkommentar neutral oder allenfalls leicht positiv verhält.
2. Sachlich-orientiertes Feedback
(Lehrer bespricht mit der ganzen Klasse die Musterlösung.)
Dieses Feedback wirkt bei allen Schülern tendenziell positiv. Aber keine bestimmte Leistungsgruppe profitiert deutlich.
3. Individuell-orientiertes Feedback
(Lehrer bespricht die einzelnen Aufgaben individuell mit dem einzelnen Schüler.)
Diese Feedbacks wirken bei allen Schülern tendenziell positiv. Aber am meisten von ihnen profitieren davon die Leistungsschwächeren.

In einer computerunterstützten Lernumgebung hat häufig der Rechner die Funktion Feedback zu geben in dem er dem Schüler Rückmeldung über dessen Aufgabenbearbeitung gibt. Bei Multiple-Choice- und Short-Answer-Aufgaben ist dies einfach, kann doch der Computer wie oben gezeigt, die Aufgabe korrigieren und ein Urteil (richtig, teilrichtig, falsch) dem Schüler abgeben (Menzel 2005).

Bei Kreativaufgaben ist dies nicht so. Da kann der Computer nicht selbstständig die Aufgaben korrigieren. Eine mögliche Art den Rechner trotzdem für Feedbacks einzusetzen, wird im folgenden Kapitel aufgezeigt. Hier wird der Computer als Transportmedium des Lehrerfeedbacks verwendet.

2.5.4 Beispiel einer Lernprozess-Implementation

Moodle ist ein sogenanntes Learning Management System (LMS), welches auch die Kreativaufgabe als Lernprozess integriert hat. Ich werde zuerst definieren was ein LMS ist, anschliessend aufzeigen wie Moodle die Kreativaufgabe eingebettet hat und wie das genau funktioniert. Obwohl Moodle ein stationäres System ist, dient dieser Ablauf der Kreativaufgabe als Vorbild für meine praktische Arbeit.

2.5.4.1 Definition von „Learning Management System“

„Als Learning Management System oder Lernplattformen werden - im Unterschied zu blossen Kollektionen von Lehrskripten und Hypertext-Sammlungen auf Web-Servern - Software-Systeme bezeichnet, die über folgende Funktionen verfügen:

- Eine Benutzeranmeldung (Anmeldung mit Verschlüsselung)
- Eine Kursverwaltung (Kurse, Verwaltung der Inhalte, Dateiverwaltung)
- Eine Rollen- und Rechtevergabe mit differenzierten Rechten
- Kommunikationsmethoden (Chat, Foren) und Werkzeuge für das Lernen (Whiteboard, Notizbuch, Annotationen, Kalender, etc.)
- Die Darstellung der Kursinhalte, Lernobjekte und Medien in einem netzwerkfähigen Browser“
(Schulmeister 2003, S. 10)

2.5.4.2 Moodle

Moodle ist eine Open Source Software, welche nach pädagogischen Prinzipien erstellt wurde um Lehrpersonen zu helfen effektive Online Learning Communities zu erstellen (Dougiamas). Speziell bei Moodle ist, dass enorm viele Plug-In's¹³ von einzelnen Usern entwickelt wurden, welche nun von jedem Administrator eingebunden werden können. Im Moment ist das Release 1.8.3 aktuell. Die offizielle Seite von Moodle ist unter www.moodle.org zu finden.



Abbildung 4: Moodle-Logo¹⁴

Ablauf einer Kreativaufgabe:

1. Die Lehrperson wählt im Drop-Down-Menu „Add an Activity“ den Eintrag „Quiz“ aus. Anschließend lädt der Browser eine Seite in dem der Lehrer die administrativen Angaben wie Zeitlimit (bis wann das Quiz gelöst werden kann), Anzahl mögliche Versuche sowie weitere Optionen eingeben kann.
2. Nachher erfasst er die Fragen für das Quiz. Es können nebst Multiple-Choice-, Short-Answer- und vielen weiteren Aufgaben auch Kreativaufgaben eingegeben werden. Der Lehrer erfasst für jede einzelne Frage den Titel, die Aufgabenstellung sowie optional die Musterlösung. Alle eingegebenen Fragen werden in einem Fragenpool gespeichert.
3. Der Schüler sieht nun auf der Plattform, dass ein neuer Test zu lösen ist. Er wählt die Aufgabenstellung an und schon wird ihm die erste Frage angezeigt. Er kann die Frage direkt im Browser beantworten: Ein Textfeld, ausgerüstet mit den wichtigsten Textbearbeitungsfunktionen, erlaubt es ihm, seine Lösung zu gestalten und die wichtigen Punkte zusätzlich hervorzuheben.
4. Nachdem er alle Fragen beantwortet hat, sendet er seine Lösung ab.
5. Der Lehrperson wird nun angezeigt, dass ein Schüler die Aufgabe gelöst hat. Sie klickt die Lösung des Lernenden an, bewertet diese Lösung und schreibt ihm die verdienten Punkte gut. Auch ist es möglich, dass die Lehrperson ein individuelles Feedback zu jeder einzelnen Lösung gibt.

Mit dieser Analyse konnte der praktische Ablauf einer Kreativaufgabe mit Computerunterstützung aufgezeigt werden. Eine sehr ähnliche Vorgehensweise wählte ich für die praktische Umsetzung dieser Diplomarbeit.

¹³ Plug-In: „Unter einem "Plug-in" (von Engl.: "to plug in"; Dt.: "hineinstecken, einstöpseln") versteht man ein Zusatzmodul, das in ein Anwendungsprogramm eingefügt wird, um dieses um zusätzliche Funktionen zu erweitern.“ <http://www.lexitron.de/main.php?detail=true&eintrag=716>

¹⁴ <http://www.theconsultants-e.com/images/photo/landscape/moodle.jpg> (08. August 2007)

3 Experteninterviews

Nachdem nun diverse theoretische Grundlagen zu den Bereichen „Lernen“ und „Spiel“ zusammengetragen wurden, geht es in den folgenden Interviews um Ideen für neue Szenarien zu sammeln und gleichzeitig allfällige begriffliche Unklarheiten aufzulösen. Das erste Interview führte ich mit meinem Betreuer Christoph Göth. Er ist verantwortlich für alle Entwicklungsschritte des mExplorers. Das zweite Interview wurde mit Dirk Frohberg durchgeführt. Er ist Spezialist im Bereich des Mobile Learnings. Er schreibt wie Christoph Göth seine Dissertation bei der Information Management Research Group¹⁵ der Universität Zürich.

3.1 Resultate aus den beiden Interviews

3.1.1 Kreativaufgaben

Es sind drei Arten von Kreativaufgaben denkbar:

1. Ein Team muss etwas gestalten: So ist es möglich, dass die Teams etwas darstellen oder etwas aufführen müssen. Dies kann das Vorführen eines Sketches sein oder eine Skulptur mit den Teammitgliedern bilden und diese fotografieren. Es kann verlangt werden etwas mit vorhandenen Materialien zu gestalten oder etwas zu basteln. Auch sind literarische Aufgaben wie das Schreiben eines Gedichtes möglich.
2. Dinge auf eine kreative Art und Weise herausfinden: Eine Beispielfragestellung lautet hier: Bestimme die Distanz zwischen Hörsaal 30 und 45 auf eine möglichst originelle Weise. Als Antworten wurde dann unter anderen 32 Knirpslängen, 3'500'000 Haarbreiten, viereinhalb Papierfliegerweiten angegeben (Göth, Frohberg und Schwabe, Von passivem zu aktivem mobilen Lernen).
3. Den physischen Kontext uminterpretieren: Clients müssen sich in einem fiktiven Szenario bewegen. So wird eine gedachte Schicht (fiktive Umgebung) über die vorhandene Realität gelegt. Zum Beispiel ist der Campus nicht mehr ein Universitätsgelände sondern ein Piratenschiff. Die Küche ist nun die Kombüse, die Cafeteria wird zum Mannschaftsraum, etc.

3.1.2 Zusammenarbeit

Es sind alle beliebigen Formen der Zusammenarbeit möglich.

1. Einzelspieler
Jeder Spieler kämpft für sich alleine und hat einen eigenen PDA.
2. Team
Ein Team bestehend aus zwei Spielern bedient einen PDA. Dieses löst gemeinsam die vorgegebenen Aufgaben. Diese Teamgrösse konnte in Untersuchungen als optimal ermittelt werden (Schwabe 2005).
3. Gruppe
Mehrere Teams werden gebildet. Anschliessend wird immer eine vorbestimmte Anzahl Teams in einer Gruppe eingeteilt.
4. Gruppen- / Team-abhängige Aufgaben
Eine Gruppe erhält eine Aufgabe, welche nur von den Teams dieser Gruppe gelöst werden sollen.

¹⁵ <http://www.ifi.uzh.ch/im> (10. August 2007)

5. Mögliche Interaktionen

Teams und Gruppen können sowohl miteinander als auch gegeneinander kämpfen. Dazu sollen die Ergebnisse der einzelnen Aufgaben untereinander aggregiert, verknüpft und ausgetauscht werden können.

So ist es möglich, dass sobald eine Aufgabe von einem Team der Gruppe gelöst wurde, nicht mehr von den Teams der anderen Gruppen bearbeitet werden kann. So entsteht regelrecht ein „Kampf“ unter den Spielern mit dem Ziel den anderen Gruppen die Tasks wegzuschnappen.

3.1.3 Lösungseingaben

Für die Beantwortung von Aufgaben stehen folgende, schon jetzt implementierte Möglichkeiten zur Verfügung:

- Textchat
- Spracheingabe
- Fotoeingabe
- Zeichnungseingabe.

Das Team soll bei der Lösungseingabe von Kreativaufgaben alle diese Eingabemöglichkeiten zur Auswahl haben.

Dirk Froberg hat noch die Möglichkeit erwähnt, dass es auch eine Lösungseingabe sein kann, wenn der Spieler sich zu einen bestimmten Ort hinbewegt und dann dort quasi seine Position als Eingabe bestätigt.

3.1.4 Bewertung der Lösung

Wenn ein Team eine Aufgabe gelöst hat, wird die Lösung über das WLAN an den Server gesendet. Es gibt nun zwei mögliche Varianten wer diese Lösung korrigieren bzw. bewerten kann:

1. Der Spielleiter

Dem Spielleiter werden die eingetroffenen Spielerlösungen angezeigt und er bewertet diese. Das Problem dieser Variante ist, dass Bewertungen von Kreativaufgaben einer gewissen Subjektivität unterliegen. Bei Aufgabenlösungen, welche sofort nach der Abgabe durch den Schüler von der Lehrperson mit der Musterlösung verglichen und als richtig, teilrichtig oder falsch bewertet werden kann, ist es angebracht, dass der Spielleiter die Korrektur vornimmt. In allen anderen Bewertungssituationen ist aber die Korrektur durch das Kollektiv (sprich den anderen Spielern) vorzuziehen, da so die Subjektivität reduziert werden kann.

2. Die anderen Spieler

Die Lösungen könnten den anderen Spielern zur Bewertung vorgelegt werden. Diese bewerten die Antworten und senden die Korrektur wieder an den Server zurück. Aus dem Durchschnitt der Bewertungen kann dann eine Rangliste über alle Lösungen erstellt werden.

Die Lösungen können mit einer Notenskala (1 bis 6 in Viertelpunkten-Schritten) oder auch mit einer Nominalskala (ist besser als) bewertet werden.

Eine weitere Möglichkeit ist die Aufgabenlösungen für eine spätere Weiterbearbeitung auf eine Webseite zu übertragen. Diese Lösungen können dann für andere Zwecke wiederverwendet werden. Dies kann interessant sein bei einer Museumsexkursion, bei der die Lernenden zurück in der Schule einen Vortrag über einzelne Ausstellungsobjekte halten müssen.

Wenn eine Aufgabe durch den Spielleiter korrigiert werden soll, ist das First-In-First-Out-Prinzip¹⁶ vorzuziehen. Es soll aber auch möglich sein, dass der Spielleiter aus einer Liste eine Lösung auswählt, die er gerade korrigieren möchte.

Als weiterer Aspekt bei der Korrektur sollte betrachtet werden: was passiert mit einer falschen Aufgabenlösung? Der Spielleiter soll die Möglichkeit haben, dem Team die Aufgabe noch einmal zurückzusenden mit dem Kommentar, dass sie falsch bzw. unvollständig sei und so den Spielern noch eine Chance zur Verbesserung ihrer Antwort geben.

3.1.5 Zeitpunkt der Bewertung und Feedback

Die Lösungen sollten bei Fragen, welche eine einfache Rückmeldung erlauben, sofort korrigiert werden. Hier ist es angebracht, dass der Spielleiter die Schülerantworten mit einer Musterlösung vergleicht.

Bei Kreativaufgaben, bei denen die Spieler etwas gestalten müssen, ist es sinnvoll das Spiel in zwei Phasen zu gliedern:

- 1. Phase: Lösen der Aufgaben
Die Teams erhalten die Aufgabe, lösen diese und senden die Antworten an den Server zurück.
- 2. Phase: Beurteilung der Antwort
Die Antworten werden an die einzelnen Spieler versandt, sobald der Spielleiter die erste Phase beendet hat. Die Teams erhalten nun die einzelnen Lösungen und bewerten diese. Sind alle Lösungen von allen Teams beurteilt, erstellt der Server eine Rangliste und versendet diese den Teilnehmern.
Zusätzlich soll es noch möglich sein, dass der Korrigierende eine Lösung oder eine Bewertung kommentieren kann. Dies ermöglicht dem Spieler die erhaltenen Bewertungen nachzuvollziehen und so einen Lerneffekt zu erzielen.

3.1.6 Weitere Anforderungen

- Zuweisung der Aufgaben
Die Aufgaben sollen den Spielern gezielt zugewiesen werden können (mögliche Varianten):
 - a. Postenlauf
Die Teams rotieren nach jedem Task zum nächsten in einer vorbestimmten Reihenfolge und nach Ablauf einer gewissen Zeit. Nachteile: Die Aufgaben müssen alle etwa gleich lang dauern, damit keine Staus entstehen. Zusätzlich sollten die Aufgaben ungefähr dieselbe Entfernung untereinander haben. Das Problem im Moment bei mobilen Learning Systemen: Muss die Technik des PDAs neu gestartet werden, wird viel Zeit benötigt und unter Umständen kann das Team dadurch seine Lösung nicht mehr eingeben.
 - b. Muss- / Soll-Aufgaben
Zuerst werden nur alle Muss-Aufgaben angezeigt. Hat das Team alle diese Aufgaben gelöst, werden die zusätzlichen Soll-Aufgaben eingeblendet. Dies hat den Vorteil, dass stärkere Teams nicht schon nach einer kurzen Zeit unbeschäftigt sind, sondern weiterhin gefordert werden können. Dies ist vergleichbar mit einem Spiel, welches mehrere Schwierigkeitsgrade hat, in welchen der starke Spieler anspruchsvollere Aufgaben lösen muss als der Schwächere.

¹⁶ First-In-First-Out-Prinzip: Die Aufgaben werden in der Reihenfolge korrigiert in der sie beim Server eingegangen sind.

c. Reihenfolge- / Kettenaufgaben:

Aufgaben, welche aufeinander aufbauen: Die erste Aufgabe muss zuerst gelöst werden, damit die nächste angezeigt wird. Die Abfolge einer logischen Verkettung wird so eingehalten.

- Eine Rangliste soll schon während des Spiels dem Client angezeigt werden können, sofern es die Aufgaben bzw. die Korrekturen zu lassen.
- Bei Kreativaufgaben sollte es möglich sein, dass der Client die eingegebene Lösung verbessern kann. So können spontane Einfälle zu einer bereits gelösten Frage immer noch eingebracht werden.
- Es sollte unbedingt möglich sein, dass der Spielleiter den Teams Hinweise gibt. Wichtig ist einfach Hilfestellungen dem Team erst zu senden, wenn es diese auch benötigt. Es sollte einen Anreiz geschaffen werden so wenig Hilfe wie möglich zu beanspruchen. Dies kann z.B. über Punkte geregelt werden: Ersucht ein Team Hilfe vom Spielleiter, erhält es automatisch Strafpunkte.
- Gemäss übereinstimmender Aussage von Christoph Göth und Dirk Frohberg sollten so um die sieben Aufgaben in 90 Minuten gestellt werden. Wobei diese Zahl vom Schwierigkeitsgrad der Aufgaben abhängig ist: werden nur Zählaufgaben gestellt, so sind in 90 Minuten 20-30 Aufgaben machbar. Es ist aber auch nur eine Aufgabenstellung denkbar, welche der Spieler 90 Minuten lang lösen muss (zum Beispiel: „Finden und bewerten Sie alle Kunstwerke von Van Gogh ...“).
- Das „Fokusproblem“ sollte bei der Zusammenstellung der Aufgaben immer beachtet werden. Dieses Theorem besagt, wenn die Spieler zu viel mit dem PDA interagieren, nehmen sie nur wenig bis nichts von der realen Spielumgebung wahr (Göth, Schwabe und Frohberg, The Focus Problem in Mobile Learning 2006).

4 Szenarien

Im Literaturteil und den Interviews konnten diverse Ansätze von möglichen Teilen von mExplorer-Szenarien festgehalten werden. Diese Teile gilt es nun zusammenzufügen, damit durchgehende Prozesse entstehen. Aufgrund dieser Analyse ist es dann möglich, die zu implementierenden Varianten auszuwählen. Ich gehe bei diesen Szenarien immer davon aus, dass diese in einer perfekten technischen Welt ohne Restriktionen umgesetzt werden können. Dies erlaubt es, alle Gedanken und Ideen zusammenzutragen ohne jetzt schon Abstriche vornehmen zu müssen.

Es wird versucht die einzelnen Möglichkeiten in Module aufzuteilen. Auf eine einfache Art können dann nach dem Baukastensystem neue Varianten zusammengesetzt werden. Diese Module werden anhand des unten abgebildeten Lernprozesses gegliedert. Gegen Ende dieses Kapitels wird die Vorgehensweise für die Bildung von Szenarien anhand der vorgestellten Prozessbausteine aufgeführt und an einem Beispiel demonstriert.



Abbildung 5: Der Lernprozess (eigene Darstellung)

4.1 Phase 1 - Aufgabenstellung

Die folgenden vier Möglichkeiten können als Prozessbausteine für die Aufgabenstellung verwendet werden. In den nachfolgenden Unterkapiteln werden diese Module genauer beschrieben.

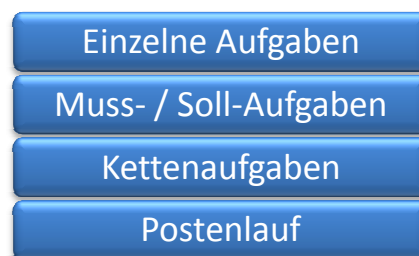


Abbildung 6: Module Phase 1 (eigene Darstellung)

4.1.1 Einzelne Aufgaben

Unter „einzelne Aufgaben“ ist zu verstehen, dass die Fragen komplett voneinander unabhängig sind, wie das bis anhin auch bei den anderen Tasks der Fall ist. Es wird also eine Frage gestellt und der Spieler muss die Antwort geben. Hat er die Antwort abgegeben, kann er direkt eine nächste, von ihm ausgewählte Aufgabe anlaufen und bearbeiten. Dieses Modul ist zwingend zu verwenden: Entweder als einzelne Aufgabenstellung oder auch um die drei anderen, nachfolgenden Module zu „füllen“. Diese bestehen jeweils aus mehreren einzelnen Aufgaben.

4.1.2 Muss- / Soll-Aufgaben

Das Lösen der Aufgaben soll in zwei Phasen aufgeteilt werden: In einer ersten Phase sind alle Aufgaben zu lösen, welche der Spielleiter als Muss-Aufgaben deklariert hat. In der zweiten Phase hingegen werden dem Spieler die Soll-Aufgaben angezeigt, die er beantworten soll, sobald er alle Muss-Aufgaben absolviert hat. Dieses zwei Phasen-Modell unterstützt leistungsstärkere Schüler, werden doch so „Bonus-Aufgaben“ gestellt, welche dem Lernenden Zusatzpunkte einbringen können und ihn so zu Mehrarbeit motivieren.

Bei der Erfassung der Aufgaben muss die Lehrperson immer unterscheiden, ob diese Frage eine Muss- oder eine Soll-Aufgabe ist. Es ist denkbar, dass einfach bei der Eingabe der Aufgabenstellung am entsprechenden Ort ein Häkchen gesetzt werden muss.

Ein Beispiel zur Veranschaulichung: Der Schüler hat Kopfrechenaufgaben zu lösen. Jeder Schüler muss das 1x1 komplett beherrschen. Wer damit fertig ist, erhält vom Lehrer komplexere Rechenaufgaben. Diese Aufgaben sollen stärkere Schüler fördern und haben den zusätzlichen Nutzen, dass ihnen nicht langweilig wird, während sie warten müssen bis die schwächeren Lernenden die Muss-Aufgaben gelöst haben.

4.1.3 Kettenaufgaben

Kettenaufgaben sind Aufgaben bei denen die nächste Frage erst dann gestellt werden kann, wenn die vorhergehende Aufgabenstellung gelöst wurde. Die Aufgaben bauen also logisch aufeinander auf. Der Schüler hat sich zuerst das Wissen der ersten Aufgabe anzueignen bevor er die zweite Frage in Angriff nehmen und lösen kann.

Die Lehrperson muss somit bei der Erstellung der Aufgaben eine Reihenfolge definieren und festlegen, welche Frage nach welcher angezeigt werden soll. Es können mehrere solcher Kettenaufgaben im System erfasst werden.

Wiederum erläutert ein Beispiel aus der Mathematik das Prinzip der Kettenaufgaben: Eine Rechnung ergibt ein erstes Resultat. Eine Kettenaufgabe verwendet dieses erste Resultat aus der vorhergehenden Aufgabe als einen Teil der neuen Aufgabe. Konkretes Beispiel: Die folgenden drei Aufgaben ergeben als Resultat 86.

1. Aufgabe: $5 \times 5 (= 25)$

2. Aufgabe: $+18 (= 43)$

3. Aufgabe: $\times 2 (= 86)$

Diese Art von Aufgaben kann auf andere Themengebiete umgeformt werden, bei denen die Abfolge wichtig ist und zusätzlich den Schüler in seinen Handlungen leiten soll.

4.1.4 Postenlauf

Bei einem Postenlauf rotieren die Teams immer in derselben Reihenfolge von einem Task zum nächsten. Haben sie einen Task gelöst, müssen sie warten bis die Zeit verstrichen ist und erst dann dürfen sie den nächsten Task ansteuern. Es sollten alle Aufgaben ungefähr gleichviel Zeit beanspruchen, damit ein „Stau“ bei zeitlich anspruchsvolleren Fragestellungen verhindert werden kann.

Der Spielleiter muss bei der Erfassung eines Postenlaufs eingeben, welche Aufgabe nach welcher zu lösen ist und auch definieren, wie lange man für das Lösen der Aufgabe Zeit hat bevor sich der Spieler zur nächsten Aufgabe begeben darf.

Ein Beispiel-Postenlauf, welcher für den Geschichtsunterricht verwendet werden kann, habe ich nachfolgend aufgeführt: Es gibt einen Rundlauf zum Thema "Schweiz" mit vier Posten für welche immer 30 Minuten zur Verfügung stehen. Es werden nun vier Aufgaben aufgeführt, welche die Schüler in diesem rund zweistündigen Postenlauf zu lösen haben:

1. Posten: „Zeigt die Gründe auf, weshalb die Eidgenossen die Schlacht am Morgarten 1315 gewinnen konnten.“

2. Posten: „Bereitet eine Parodie über Schillers ‚Wilhelm Tell‘ vor. Diese ist anschliessend an diesen Postenlauf vor versammelter Klasse vorzuspielen.“
3. Posten: „Beantwortet das aufliegende Fragenblatt zu General Guisans Redit-Idee.“
4. Posten: „Erstellt eine Kurzübersicht über die wichtigsten Positionen der vier Bundesratsparteien.“

4.2 Phase 2 - Lösen der Aufgabe

Aus den vorhergehenden Kapiteln lassen sich drei mögliche Module für das Lösen der Aufgaben identifizieren:

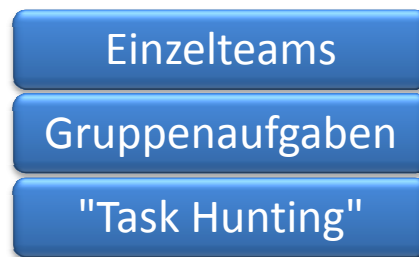


Abbildung 7: Module Phase 2 (eigene Darstellung)

Das Gemeinsame bei allen diesen Modulen ist die Art wie der Spieler die Lösung ins System eingeben kann:

- Texteingabe:
Der Lernende gibt seine Lösungen in ein Textfeld ein und schickt anschliessend die Antwort dem Spielleiter.
- Foto-Annotation:
Der Lernende macht ein Bild mit der im PDA integrierten Kamera und versendet dieses.
- Voice-Annotation:
Der Lernende nimmt ein Memo über das im PDA integrierte Mikrofon auf. Der Spielleiter kann über die mExplorer-Steuerungssoftware die Lösung des Lernenden abhören und dann bewerten.
- Handschrift-Annotation:
Der Lernende zeichnet die Lösung auf und sendet diese dem Spielleiter.

Die verschiedenen Annotationen wurden von Christoph Wirz in seiner Diplomarbeit „*Annotationen im Mobile Learning Game*“ in den mExplorer implementiert. Es sollte also möglich sein diese Programmteile zu übernehmen und für die Beantwortung von Kreativaufgaben zu verwenden.

4.2.1 Einzelteams

Ein Team (zwei Personen, ein PDA) löst die Tasks unabhängig von den anderen Teams. Diese Möglichkeit ist die bisher einzige Variante, welche im mExplorer schon vollständig umgesetzt und verwendet wurde.

4.2.2 Gruppenaufgaben

Gruppenaufgaben sind Aufgaben bei denen sich einzelne Zweierteams zusammenschliessen und gemeinsam eine Aufgabe lösen. Der Spielleiter soll bei der Erstellung der Gruppenaufgaben ein entsprechendes Häkchen setzen. Hat eine Gruppe diese Aufgabe gelöst, verschwindet das Aufgabensymbol auf der Karte jedes einzelnen Gruppenmitglieds.

4.2.3 „Task Hunting“

Mit Task Hunting ist die Möglichkeit gemeint, dass ein Team den anderen Teams Aufgaben wegschnappen kann: Hat ein Team eine Aufgabe richtig gelöst, wird dieser Task den anderen Teams nicht mehr angezeigt. Es entsteht so quasi ein Wettbewerb wer die meisten Tasks lösen kann. Es ist vorstellbar, dass dieses Szenario als Teil eines Spiels (mit den sechs Spielfaktoren von Prensky) in den mExplorer eingebettet wird. Die grösste Schwierigkeit bei diesem Modul ist die Korrektur von Kreativaufgaben, welche eine gewisse Zeit beansprucht und dadurch die Lösung nicht unmittelbar nach dem Absenden als richtig oder falsch bewertet werden kann. Dies hat zur Folge, dass ein Task nicht sofort für die anderen Teams gesperrt wird, sobald er richtig gelöst wurde. Es wäre in diesem Fall möglich, dass mehrere Teams die Aufgabe richtig lösen, was dem Spiel die „Würze“ nehmen und das Endresultat verfälschen würde.

Es muss zusätzlich folgende Einschränkung beim Modul „Task Hunting“ gemacht werden: Wird dieses Modul ausgewählt, kann sinnvollerweise aus der Phase 1 nur noch das Modul „einzelne Aufgaben“ verwendet werden. Alle anderen Module würden sonst in ihrem Nutzen zerstört werden.

4.3 Phase 3 - Korrektur / Bewertung

Bei der Korrektur und Bewertung gibt es zwei Sets mit Modulen. Das erste Set beschreibt wer alles korrigieren kann, während das zweite vorgibt mit welchem Schema die Bewertung durchgeführt wird.



Abbildung 8: Module Phase 3 - 1 (eigene Darstellung)

4.3.1 Spielleiter

Der Spielleiter ist derjenige, welche das Spiel an einem Computer überwacht. Wenn nun ein Team eine Aufgabe gelöst hat und die Antwort dem Spielleiter sendet, wird diese in einer Liste gespeichert sowie auf dem Bildschirm des Spielleiters eine Nachricht angezeigt, dass eine neue Antwort zum Korrigieren vorliegt. Der Spielleiter wählt die neue Antwort aus und korrigiert diese anhand eines bei der Aufgabenstellung definierten Bewertungsschemas. Nachdem die Bewertung vollzogen ist, wird das Resultat gespeichert und je nach Feedbackvariante dem Spieler sofort mitgeteilt oder erst verzögert kommuniziert.

4.3.2 Andere Spieler

Wenn die Aufgaben so gestellt werden, dass die anderen Spieler die Lösungen bewerten sollen, ist es angebracht das Spiel in die bekannten zwei Phasen aufzuteilen. Für die Korrektur werden die Aufgaben vom Server an die Spieler gesendet, der Spieler nimmt die Bewertungen vor und schickt das Urteil zurück an den Server, welcher mit den neuen Korrekturen eine Rangliste erstellt.

Konkret könnte dies so aussehen, dass dem Spieler jeweils zwei Lösungen angezeigt werden, von denen er diejenige anklicken muss, welche ihm besser gefällt. Dies wird so oft wiederholt, bis jede Lösung mit jeder anderen einmal verglichen wurde und eine vollständige Rangliste erstellt werden kann.

Welche Möglichkeiten gibt es um Lösungen der Spieler zu beurteilen? Im Literaturteil wurden drei unterschiedliche Bewertungsschemata vorgestellt. Bei jeder einzelnen Kreativaufgabe soll der Spielleiter entscheiden können, welches Schema er für die neue Aufgabe verwenden will.

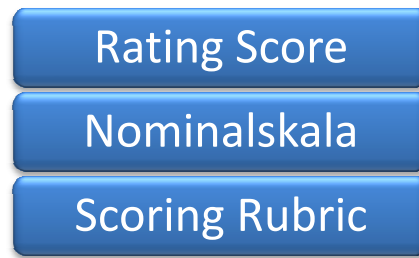


Abbildung 9: Module Phase 3 - 2 (eigene Darstellung)

4.3.3 Rating Score

Die Lehrperson soll innerhalb der Spielregeln eine Maximalpunktzahl eingeben können. Diese darf bei der Bewertung der Spielerlösungen nicht überschritten werden.

Trifft beim Server eine Lösung ein, öffnet der Korrigierende diese. Das Fenster wird nun vertikal geteilt: Auf der einen Seite wird die Lösung des Spielers angezeigt und auf der anderen die Aufgabenstellung und die Musterlösung. So wird es dem Spielleiter vereinfacht die Korrektur vorzunehmen, die Punkte zu verteilen und allenfalls eine Bemerkung (Feedback) anzubringen.

Diese Bewertungsmethode mit Musterlösung ist dann am sinnvollsten wenn der Spielleiter offene Fragen stellt, um Wissen und Verständnis zu testen, in denen keine grossen Kreativleistungen des Spielers notwendig sind. Dies ist vergleichbar mit einer klassischen schriftlichen Prüfung, welche auch objektiv korrigiert werden kann.

Ein Beispiel aus der Betriebswirtschaftslehre: Auf die Aufgabe "Nenne drei Eigenschaften der Matrix-Organisationsstruktur" antwortete der Spieler wie folgt:

1. Es werden zwei Gliederungsprinzipien hierarchisch gleichberechtigt verwendet.
2. Die Stabsstelle ist eine Leitungshilfe für die Entscheidungsträger.
3. Entscheidungsverfahren „Selbstabstimmung“ wird bewusst in der Struktur verankert.

Der Spielleiter korrigiert diese Aufzählung anhand der Musterlösung. Da der Spieler nur zwei richtige Eigenschaften (1. & 3.) auführte, erhält er nur zwei Drittel der maximalen Punktzahl.

4.3.4 Nominalskala

Mit dieser Bewertungsmethode werden Vergleiche zwischen zwei Lösungen angestellt. Dies könnte folgendermassen programmiert werden: Der Server zeigt dem Korrigierenden zwei Lösungen in einem wiederum vertikal geteilten Fenster an. Der Korrigierende erstellt nun eine absteigende Reihenfolge der einzelnen Lösungen. Die beste Lösung hat den obersten Platz während die schlechteste den untersten einnimmt. Es ist denkbar, dem Korrektor die Aufgaben anonym vorzulegen, damit die Objektivität noch weiter erhöht wird.

Mit all diesen Bewertungen kann von einer einzelnen Schülerlösung ein Prozentsatz berechnet werden, der besagt, dass die vorliegende Lösung in X-Vergleichen mit anderen Lösungen, Y-mal den Vorzug erhalten hat, also „als besser“ gewählt wurde. Die Lösung, welche den höchsten Prozentsatz aufweist, wird auch in der Rangliste am höchsten positioniert.

4.3.5 Scoring Rubric

Bei der Erstellung der Fragen wählt der Spielleiter als Korrekturmethode die Scoring-Rubric-Variante aus. Er erstellt diese Tabelle nach dem Erfassen der Aufgabe und speichert sie ab. Auch mit diesem Modul empfiehlt es sich, sofern die anderen Spieler die Korrigierenden sind, das Spiel in zwei Phasen zu untergliedern. In der zweiten Phase wird dann vom Server immer eine Lösung mit einer leeren Scoring-Rubric-Tabelle an den Korrektor gesendet. Der Korrigierende liest die Lösung durch und wählt auf der Tabelle die entsprechenden Zellen an. Hat der Server von allen Teams die Korrekturen erhalten, berechnet er von jeder einzelnen Korrektur die erreichten Punkte und rechnet über alle Bewertungen den Durchschnitt aus. Je höher der Durchschnitt im Vergleich zu den anderen Spielerlösungen, desto besser wird der Spieler klassiert.

Ein Beispiel wurde bereits im theoretischen Teil aufgezeigt.

4.4 Phase 4 - Feedback

In der letzten Phase wird dem Spieler die Bewertung seiner Lösung angezeigt. Die Feedback-Phase kann aus den zwei folgenden Modulen bestehen:



Abbildung 10: Module Phase 4 (eigene Darstellung)

4.4.1 Rangliste

Wenn eine Rangliste erstellt werden kann und auch ein gewisser Wettbewerbsgedanke unter den einzelnen Teams herrscht, ist es für die Spieler interessant zu sehen wie gut sie im Vergleich zu den anderen abgeschnitten haben. Eine Rangliste ist immer auch eine High-Score-Liste. Je besser man ist, desto weiter oben wird man platziert. Wie in der theoretischen Einführung aufgezeigt werden konnte, motiviert das Ziel den Menschen („möglichst hohen High-Score-Listeneintrag erzielen“). Darum ist die Rangliste ein Motivationsfaktor für die Spieler auf den nicht verzichtet werden sollte.

Die Rangliste kann erst am Ende angezeigt werden. Es ist aber auch möglich, dass (sofern der Spielleiter die Aufgaben während dem Spiel korrigiert) der Punktezweischenstand bereits während dem Spiel von den Teams abgerufen werden kann.

4.4.2 Feedback

In der theoretischen Einführung wurde erwähnt, dass das Feedback dem Lernenden mitteilt, was er falsch machte bzw. was die richtige Lösung gewesen wäre. Daraus zieht der Schüler Konsequenzen und passt sein Verhalten und sein Wissen dementsprechend an.

Aus diesem Grund ist es wichtig Feedback über die Leistungen des Lernenden zu geben. Am besten eignet sich eine kurze Erklärung, weshalb er mit seiner Antwort nicht die volle Punktzahl erreichte und was er hätte schreiben müssen, damit die Lösung besser bewertet worden wäre.

Auch ist es möglich dem Schüler die Musterlösung anzuzeigen, damit er selber erkennen kann, wo seine Fehler liegen. Eine Kombination der beiden Varianten ist durchaus realistisch und vermindert der Lehrperson den Aufwand zu viel „Standard-Kommentar“ zu den einzelnen Lösungen schreiben zu müssen.

Es wird zwischen unmittelbarem und zeitlich verzögertem Feedback unterschieden: Beim unmittelbaren Feedback wird sofort nach dem Absenden der Antwort die Musterlösung angezeigt. Diese Variante kann den Nachteil haben, dass der Spieler in seinem „Spieltrieb“ das Feedback einfach wegklickt, da er sich schon auf dem Weg zur nächsten Aufgabe befindet und ihn im Moment die Musterlösung wenig interessiert. Beim zeitlich verzögerten Feedback erhält der Spieler den Kommentar erst, wenn die Spielphase sowie der Korrekturteil abgeschlossen sind. Das Problem des Wegklickens kann so umgangen werden.

4.5 Bildung der Szenarien

Bevor der Spielleiter daran geht, die einzelnen Aufgaben im System zu formulieren, muss er sich zwingend über die Zusammenarbeit der Teams Gedanken machen und ein entsprechendes Modul der Phase 2 auswählen. Anschliessend können dann die Module der anderen Phasen zusammengesetzt werden. Zum Schluss hat der Spielleiter noch die Möglichkeit zu bestimmen, ob die Module der vierten Phase verwendet werden sollen oder nicht. Anhand des nachfolgenden „Beispiel-Szenarios“ werde ich die Verwendung dieser oben erwähnten Module systematisch aufzeigen.

4.5.1 Beispiel-Szenario

Als erstes definiere ich, dass die Teams einzeln für sich die Aufgaben lösen müssen (Modul: *Einzelteams*). Ich möchte also nicht, dass die Teams zusammenarbeiten. Es soll ein Wettbewerb zwischen den einzelnen Teams entstehen.

Nun geht es um die Wahl der Aufgabenart. Ich wähle hier das Modul *Muss-/Soll-Aufgaben*. Es gibt also Aufgaben, welche zwingend von allen gelöst werden sollen und zusätzliche, welche von den schnelleren Teams noch gelöst werden können und Extrapunkte geben.

Jetzt kann ich die Aufgaben erfassen: Ich erstelle die erste Kreativaufgabe (Modul: *einzelne Aufgabe*). Als Erstes lege ich einen Titel fest, gebe die Frage ein und setze bei „Muss-Aufgabe“ ein Häkchen.

Anschliessend definiere ich, dass die Antwort von den *anderen Spielern* (Modul von Phase 3) korrigiert werden soll, und zwar mit einer *Nominalskala*, bei der die Lösungen mit „ist-besser-als“ verglichen werden.

Für jede Aufgabe wiederhole ich diese drei Auswahlen (Muss-/Soll-Aufgabe; Korrigierender; Bewertungsskala).

Habe ich nun alle Aufgaben im System erfasst, gehe ich über zu den Modulen der vierten Phase und selektiere die gewünschten Eigenschaften:

- *Rangliste*

Da das Ganze in einem Spielrahmen ablaufen soll, möchte ich, dass die Spieler sehen, wie gut sie im Vergleich zu den anderen Teams die Aufgaben gelöst haben und aktiviere somit die Rangliste.

- *Feedback*

Damit die Lernenden ihre Fehler kein zweites Mal begehen, bzw. Erkenntnisse für die Zukunft ziehen können, ist es angebracht, ihnen ein zeitverzögertes individuelles Feedback inklusive Musterlösung zu geben. Dieses Feedback wird ganz am Ende des Spiels den Spielern verteilt.

Mit diesem Beispiel-Szenario konnte die Vorgehensweise bei der Erstellung eines Szenarios anhand der vorliegenden Module aufgezeigt werden. Die ganze Zusammensetzung solcher Szenarien ist sehr intuitiv und bereitet nur wenig Mühe.

4.6 Analyse der einzelnen Module

Damit entschieden werden kann, welche Module im mExplorer umgesetzt werden können, ist eine Stärken-Schwächen-Analyse notwendig. Vor allem sollen nun auch die technischen Grenzen berücksichtigt werden.

4.6.1 Module der Phase 1

Modul	Stärke / Schwäche
Einzelne Aufgaben	<ul style="list-style-type: none"> + Wird als „Universalmodul“ von allen anderen Modulen der Phase 1 als Basis verwendet und muss nur für den spezifischen Verwendungszweck angepasst werden.
Muss-/Soll-Aufgaben	<ul style="list-style-type: none"> + Stärkere Schüler werden gefördert. + Technisch muss nur überprüft werden, ob schon alle Pflichtaufgaben gelöst wurden bevor die Soll-Aufgaben angezeigt werden. - Schwächere Schüler werden nicht speziell gefördert.
Kettenaufgaben	<ul style="list-style-type: none"> + Es werden nur die Fragen angezeigt, welche Startpunkte oder nachfolgende Fragen sind. - Fragestellungen sollten aufeinander aufbauen.
Postenlauf	<ul style="list-style-type: none"> + Spieler vertrödeln keine Zeit beim Lösen einer Aufgabe aufgrund des vorhandenen Zeitdrucks. - Alle Aufgaben müssen ungefähr gleichlange dauern. - Bei einer instabilen Technik kann die Zeitvorgabe zum Demotivationsgrund für die Spieler werden, wenn das Gerät neu gestartet werden muss und zwischenzeitlich die Eingabezeit verstrichen ist.

Tabelle 7: Übersicht Stärken / Schwächen der Module der Phase 1

4.6.2 Module der Phase 2

Modul	Stärke / Schwäche
Einzelteams	<ul style="list-style-type: none"> + Wird jetzt schon so angewendet. - Erzwingt keine Kollaboration der Teams über das mobile Gerät.
Gruppenaufgaben	<ul style="list-style-type: none"> + Fördert die Zusammenarbeit. - Interaktionen vermutlich schwierig, da das Gerät noch häufig abstürzt.
„Task Hunting“	<ul style="list-style-type: none"> + Geschwindigkeitsfaktor wird eingebaut. - Schwierig anzuwenden bei Kreativaufgaben aufgrund der verzögerten Korrektur.

Tabelle 8: Übersicht Stärken / Schwächen der Module der Phase 2

4.6.3 Module der Phase 3

Modul	Stärke / Schwäche
Spielleiter	<ul style="list-style-type: none"> + Korrektur findet schon während des Spiels statt. + Nur der Server muss mit einem Korrektur-Programmteil ausgestattet werden.
andere Spieler	<ul style="list-style-type: none"> - Subjektivität in der Beurteilung der Lösung. + Korrektur wird objektiviert.
Rating Score	<ul style="list-style-type: none"> - Spiel muss in zwei Phasen aufgeteilt werden, damit die Spieler die Bewertungen seriös durchführen. + Einfaches Schema v.a. wenn die Lösung nur mit der Musterlösung verglichen werden muss. + Vor allem für offene Wissens- oder Verständnisfragestellungen geeignet.
Nominalskala	<ul style="list-style-type: none"> - Musterlösung sollte dem Korrigierenden während der Korrektur angezeigt werden um Unregelmässigkeiten zu vermeiden. + Vor allem für Kreativaufgaben geeignet.
Scoring Rubric	<ul style="list-style-type: none"> - Aus Gründen der Fairness sollten alle Lösungen gleichhäufig mit anderen Lösungen verglichen werden. + Objektivität der Korrektur wird gross geschrieben. - Aufwendige Erstellung: Bei vielen unterschiedlichen Inhalten sind viele Scoring Rubrics zu erstellen. Die Lehrperson wird also diese Bewertungsvariante versuchen zu umgehen.

Tabelle 9: Übersicht Stärken / Schwächen der Module der Phase 3

4.6.4 Module der Phase 4

Modul	Stärke / Schwäche
Rangliste	<ul style="list-style-type: none"> + Gibt Spielern eine gewisse Genugtuung. - Nicht immer sind Ranglisten sinnvoll. Sie können auch demotivieren.
Feedback	<ul style="list-style-type: none"> + Gibt dem Spieler individuelle Rückmeldung. - Muss sinnvoll gestaltet werden, damit es praktisch ist zum Lesen auf dem kleinen Bildschirm.

Tabelle 10: Übersicht Stärken / Schwächen der Module der Phase 4

5 Implementation

In diesem Hauptteil wird die Realisierung der neuen Komponenten beschrieben. Zuerst wird die Auswahl an Modulen vorgestellt, welche dem mExplorer hinzugefügt wurden. Anschliessend wird für jedes Modul der technische Ablauf inklusive den dazugehörenden Benutzerinteraktionen aufgezeigt.

5.1 Modulauswahl / Anforderungsspezifikation

Von grosser Wichtigkeit in einem Softwareprojekt ist die Anforderungsspezifikation. Sie beinhaltet eine Zusammenstellung der Fähigkeiten, welche die Software haben muss (Glinz WS 2005/2006). In einer Sitzung mit Christoph Göth wurde entschieden, dass die folgenden Module neu im mExplorer einzubetten sind. Bereits implementierte Komponenten werden hier der Vollständigkeit halber einfach erwähnt um den oben vorgestellten Lernprozess komplett zeigen zu können.

Phase 1 - Aufgabenstellung

- Einzelne Aufgaben

Die einzelnen Aufgaben sind insofern schon im mExplorer enthalten, da geschlossene Aufgabentypen wie Multiple-Choice- und Short-Answer-Aufgaben schon programmiert sind. Mit diesem Modul soll neu der Aufgabentyp „Kreativaufgaben“ implementiert werden. So sollen in Zukunft auch vollständig offene Fragen durch den Spielleiter gestellt werden können.

- Kettenaufgaben

Es soll eine Art „Container“ entwickelt werden, in den einzelne Aufgaben platziert werden können. Dieser Container stellt dann die Einhaltung der richtigen Reihenfolge der einzelnen Aufgaben sicher. Auf dem Bildschirm des Spielers darf nur eine Aufgabe angezeigt werden und erst wenn er diese gelöst hat, soll die Nachfolgende erscheinen. Das Lösen von Kettenaufgaben ergibt aber keine zusätzlichen Punkte. Nur für die einzelnen Teilaufgaben werden Punkte verteilt.

Phase 2 - Lösen der Aufgabe

- Einzelteams

Diese Art des Aufgabenlösen wird bereits seit der prototypischen Implementierung des mExplorers verwendet. Sie muss nicht ausgebaut werden.

- Gruppenaufgaben

Bei der Erstellung von Kreativaufgaben soll der Spielleiter beim Erfassen der Aufgabe anwählen können, ob die Aufgabe alleine oder innerhalb der Gruppe zu lösen ist. Beim Spieler erscheint auf dem Bildschirm die Möglichkeit anzukreuzen, mit welchen anderen Teams die Aufgabe gelöst wird. Bei der anschliessenden Korrektur sollen natürlich alle Beteiligten die gleiche Anzahl Punkte erhalten. (Die Zuweisung eines Teams zu einer Gruppe ist bereits vor dieser Diplomarbeit implementiert worden und muss nicht neu programmiert werden.)

Phase 3 - Korrektur / Bewertung

- Spielleiter

Der Zuständigkeitsbereich des Spielleiters soll erweitert werden. Er soll zusätzlich zu der Erfassung der Aufgaben auch die Antworten der einzelnen Teams und Gruppen bewerten.

- Rating Score

Beim Erfassen der Kreativaufgabe soll ausgewählt werden können, ob die Antworten mit dem Rating Score- oder dem Nominalskala-Verfahren korrigiert werden. Ebenso soll eine für alle Aufgaben geltende maximal erreichbare Punktzahl definiert werden, welche der Spielleiter bei der Kor-

rektur nicht überschreiten darf. Die einzelnen Antworten einer Aufgabe sollen übersichtlich dargestellt werden, dass immer klar ist zu welcher Aufgabe und zu welchem Team die vorliegende Antwort gehört.

- Nominalskala

Der Spielleiter soll die Antworten miteinander vergleichen können und sich dann auf eine Reihenfolge festlegen, damit die Punkte vom System berechnet werden können. Der beste Spieler erhält die maximale Punktzahl während der Schlechteste null Punkte erhält. Die Teams dazwischen erhalten ihre Punkte der Reihenfolge entsprechend linear absteigend.

Phase 4 - Feedback

- Rangliste

Es sollen alle Punkte eines Teams in einer Rangliste dargestellt werden, damit der Spielleiter jederzeit in der Spielübersicht ablesen kann, welches Team welche Rangierung inne hat.

- Feedback

Der Spielleiter muss die Möglichkeit haben bei der Aufgabenkorrektur einen Kommentar zu erfassen. Die Begründungen sind aber erst nachdem Spiel relevant: Für die Nachbesprechung des Erlebten sollen diese Kommentare verwendet werden können. Es ist daher nicht nötig die Spielleiterkommentare auf die PDAs zu übertragen. Die Rangliste mit den verteilten Punkten sollte hingegen von den Spielern jederzeit abgerufen werden können.

Zusätzliche Funktionen

Die folgenden Funktionen sollen zusätzlich implementiert werden. Sie sind entweder administrativer Natur oder übergreifen mehrere Phasen des Lernprozesses.

- Extrapunkte

Der Spielleiter soll nach seinem Erachten einzelnen Teams Extrapunkte zuweisen können. Er kann so herausragende Arbeiten zusätzlich belohnen. Diese Punkte sollen zu den anderen hinzugezählt werden und in die Gesamtangliste einfließen.

- Andere Antworten anzeigen

Möchte ein Team eine Kreativaufgabe lösen, soll es die Lösungen der anderen Teams, welche diese Aufgabe schon gelöst haben, abrufen können und sich von diesen inspirieren lassen.

Hat dann das Team die eigene Antwort abgesendet, soll es auf der Karte an die Position der gelösten Aufgabe klicken und weiterhin alle abgegebenen Antworten betrachten können, auch die der nachfolgenden Teams. Dies soll den Wettbewerb zwischen den Teams anregen und die einzelnen Antworten im Verlaufe des Spiels innovativer werden lassen.

- HTML-Export

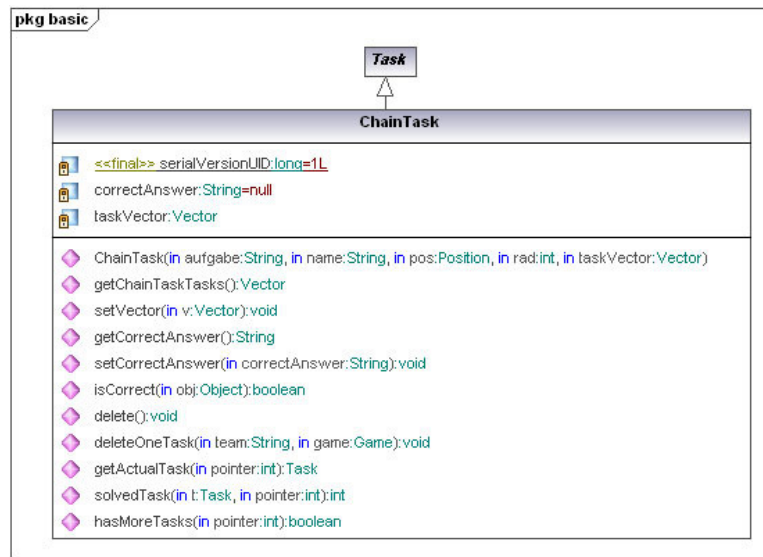
Um eine Nachbesprechung der abgegebenen Kreativaufgaben zu ermöglichen soll die bestehende Export-Funktion erweitert werden, damit die Antworten der Spieler beispielsweise im Klassenzimmer über einen Projektor angezeigt und diskutiert werden können. Die einzelnen Kreativaufgaben sind in einem internetfähigen Dokument mit den Lösungen der Lernenden darzustellen.

5.2 Umsetzung der einzelnen Module

5.2.1 Aufgabenstellung - Kettenaufgaben

Die Kettenaufgabe (*ChainTask.java*) wird als Unterklasse der *Task*-Klasse im mExplorer eingebettet. Dies kann der Abbildung 11 entnommen werden. Der Name der Kettenaufgabe („Name of the Task?“ in Abbildung 12) muss zwingend einmalig sein, damit keine Verwechslungen entstehen können. Eine Ketten-

aufgabe besteht immer aus mindestens zwei anderen beliebigen Aufgaben. Hat der Spielleiter weniger als diese zwei Aufgaben im System erfasst, kann er keine Kettenaufgabe erstellen. Die Klasse „Task“ wird angepasst, dass neu ihre Objekte auch nach Typ unterschieden werden können. Mit einer Integer-Variable (Zahlenwert) wird dieser Wert gespeichert: Hat diese Variable den Wert ‚2‘ handelt es sich um eine Kettenaufgabe. Dies ermöglicht es, dass sowohl der Server als auch der PDA überprüfen können, ob eine Kettenaufgabe vorliegt. Die *Task*-Klasse wurde noch weiter verändert, so wird über eine Boolean-Variable (Wahrheitswert) eine ‚normale‘ Aufgabe als Teil einer Kettenaufgabe deklariert. Der Name der dazugehörenden Kettenaufgabe wird in diesem Fall in einer zusätzlichen String-Variable (Textwert) gespeichert.

Abbildung 11: UML-Klassendiagramm: *ChainTask.java*

5.2.1.1 Erstellung einer Kettenaufgabe

Mit dem folgenden Dialogfenster kann eine Kettenaufgabe im Spieleditor erfasst werden:

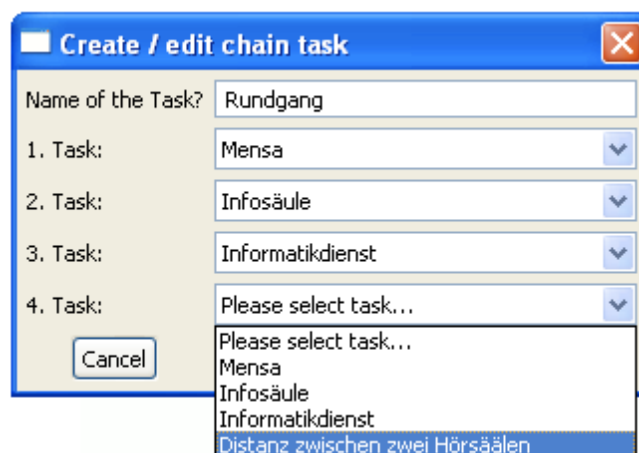


Abbildung 12: Erfassung einer Kettenaufgabe

Bei der Erstellung können alle erfassten Aufgaben (einfache oder Kreativaufgaben) ausgewählt werden, welche noch zu keiner anderen Kettenaufgabe hinzugefügt wurden. Mit einem Klick auf den [OK]-Button werden die Namen der ausgewählten Tasks in einem Vektor innerhalb der Kettenaufgabe gespeichert. Zusätzlich wird überprüft, dass keine Aufgabe zweimal selektiert wurde.

Da die Kettenaufgabe keine eigenständige Aufgabe ist, hat sie auch kein eigenes Symbol auf der Spielkarte. Ihre Teilaufgaben werden hingegen mit roten Symbolen dargestellt und mit einer Linie verbunden. Die Zahl rechts von den Symbolen gibt an, in welcher Reihenfolge die Aufgaben gelöst werden müssen.

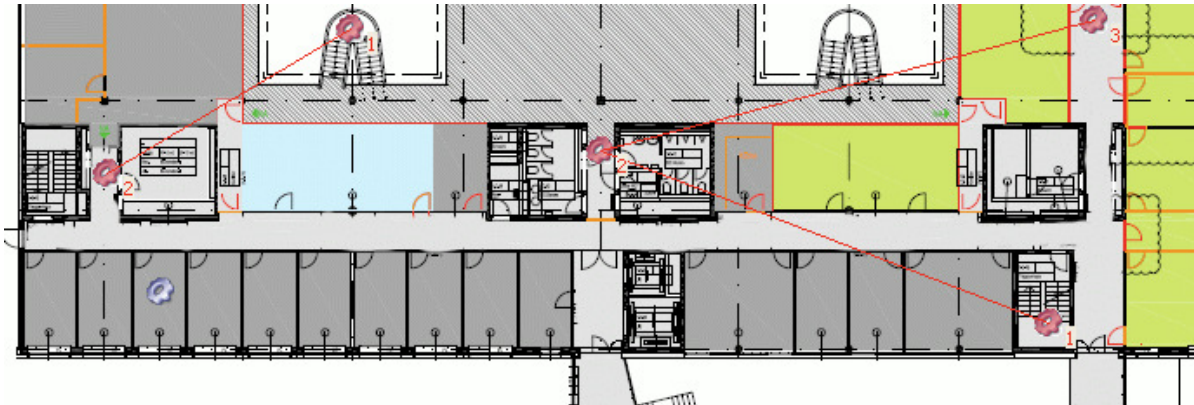


Abbildung 13: Anzeige von Kettenaufgaben im Spieleditor

Hat der Spielleiter Kettenaufgaben erstellt, darf man nicht einfach Tasks löschen, sondern der Server muss immer auch Abhängigkeiten überprüfen und entsprechend anpassen. Es sind zwei mögliche Löschanvarianten zu unterscheiden:

1. Kettenaufgabe wird gelöscht:
Alle Tasks müssen angepasst werden: Die Boolean-Variable, welche besagt, dass die Aufgabe Teil einer Kettenaufgabe ist und der im Task gespeicherte Namen der Kettenaufgabe müssen auf FALSE bzw. NULL gesetzt werden. In der Klasse *ChainTask.java* werden diese Aktionen mit der Methode *delete()* durchgeführt (siehe auch Abbildung 11).
2. Normale Aufgabe wird gelöscht:
Diese gelöschte Aufgabe ist Teil einer Kettenaufgabe. Der Kettenaufgaben-Vektor mit den gespeicherten Tasks muss demzufolge angepasst werden: Der Task wird aus dem Vektor entfernt und die Einträge mit höherer Nummer müssen eins „nach unten rutschen“. Die Kettenaufgabe wird mit der Methode *deleteOneTask(..)* aktualisiert.

5.2.1.2 Kettenaufgaben auf dem PDA

Auf der Spielkarte wird immer nur eine Teilaufgabe der Kettenaufgabe angezeigt. So kann der Spieler „gesteuert“ werden, in dem die genaue Abfolge der Kettenaufgabe eingehalten wird wie sie der Spielleiter erfasst hat. Damit ein Client weiss welche Aufgabe er im Moment anzeigen muss, enthält ein Pointer (Zeiger) die Nummer des aktuellen Tasks. Wird eine Aufgabe gelöst, erhöht sich der Pointer um eins und die nächste Aufgabe wird angezeigt. Wurde der letzte Task der Kettenaufgabe absolviert, setzt das System den Pointer auf ‚-1‘. Der PDA erkennt nun, dass keine weiteren Aufgaben vorhanden sind und zeigt ein Fenster an, in welchem die Mitteilung „Alle Tasks der Kettenaufgabe XY gelöst.“ steht.

Wenn der Spieler auf der Karte mehrere Tasks sieht, kann er nicht unterscheiden, welche Teil einer Kettenaufgabe sind und welche nicht. Es wird immer dasselbe Symbol verwendet um eine Aufgabe darzustellen. In der Aufgabenstellung wird hingegen ein Text eingeblendet, welcher dem Spieler mitteilt, dass die aktuelle Aufgabe Teil einer Kettenaufgabe ist. Mit der Nachricht „(4. von 5 Aufgaben)“ kann der Spieler selber erkennen, wie viele Tasks dieser Kettenaufgabe er schon gelöst hat bzw. wie viele ihm noch bevorstehen. Wie dies dann praktisch aussieht kann der folgenden Abbildung entnommen werden.

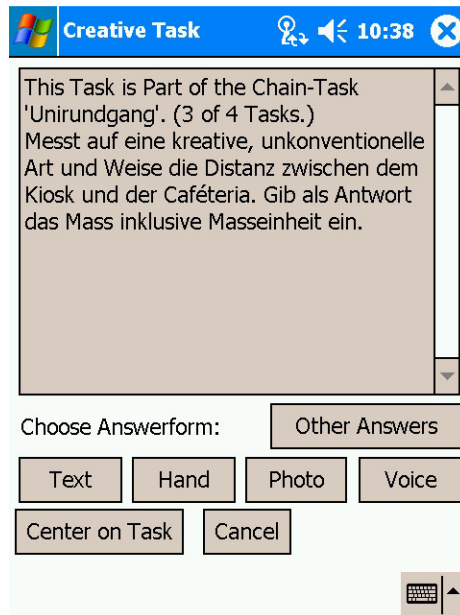


Abbildung 14: Anzeige eines Tasks einer Kettenaufgabe auf dem Client

5.2.1.3 Speichern des Pointers

Wie oben erwähnt wird der Pointer einer Kettenaufgabe auf dem PDA um eins erhöht, wenn eine Aufgabe gelöst wurde. Da nach dem Lösen eines Tasks standardmässig eine Mitteilung an den Server gesendet wird, kann der Pointer auch beim zentralen Rechner um eins erhöht werden. Dies ist notwendig, ist es doch möglich dass die Software des PDAs einfriert und der Client neu gestartet werden muss. Jetzt möchte der Spieler nicht mehr alle vorhin gelösten Aufgaben erneut absolvieren, sondern direkt dort einsteigen, wo sein Rechner abstürzte. Dies ist jedoch nur möglich, wenn der Pointer auch zentral gespeichert und beim Neustart dem Client zugeschickt wird.

Der Server speichert daher den Pointer im Status jedes Teams ab (*TeamStatus.java*). Dazu wird eine Hash-tabelle generiert, welche den Namen der Kettenaufgabe als Schlüssel verwendet und als Objekt den aktuellen Pointer speichert. So ist es auf eine einfache Art möglich alle aktuellen Pointer der verschiedenen Kettenaufgaben eines Teams zentral zu verwalten. Die Pointer werden bewusst nicht in der Klasse *Chain-Task.java* verwaltet, da dies der Software-Architektur widersprechen würde. Im mExplorer sind statische Daten im Package *data.basic* gespeichert und die dynamischen, temporären Daten werden im Package *data.game* verwaltet (in Anlehnung an die Diplomarbeit von Dennis Brunner).

Wie der Server gelöste Teilaufgaben von Kettenaufgaben verwaltet, ist im folgenden Sequenzdiagramm dargestellt und wird im Begleittext genauer erläutert.

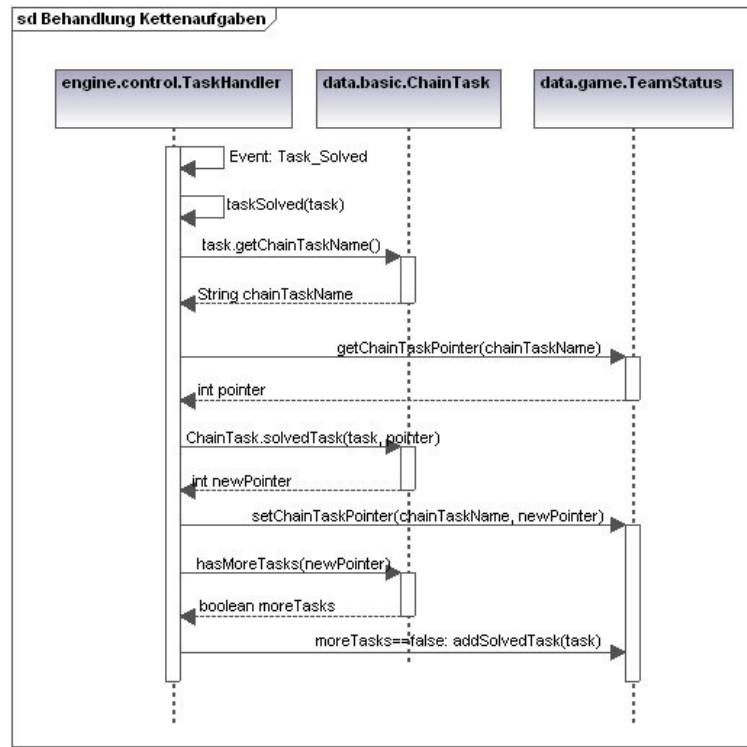


Abbildung 15: Sequenzdiagramm: Serverseitige Behandlung von Kettenaufgaben

Erhält der Server über die Netzwerkschicht ein *GameEvents.TASK_SOLVED*-Ereignis, wird dieses an den *TaskHandler* in der Logikschicht weiter gereicht. Dieser behandelt das neue Ereignis in der Methode *taskSolved(..)*. Nachdem geprüft wurde, ob diese Aufgabe Teil einer Kettenaufgabe ist, wird der eindeutige Name der Kettenaufgabe abgefragt. Mit diesem Namen ruft der *TaskHandler* beim *TeamStatus* des Senderteams den aktuellen Pointer ab. Mit der zu Beginn empfangenen Aufgabe und dem aktuellen Pointer wird die Kettenaufgabe aufgerufen und der Pointer um eins erhöht. Dieser neue Pointer wird anschließend wieder im *TeamStatus* gespeichert. Dadurch hat der Server wieder den aktuellen Pointer in der Hash-tabelle. Zum Schluss wird überprüft, ob die gelöste Teilaufgabe die letzte der Kettenaufgabe war oder nicht. Sind keine weiteren Aufgaben vorhanden, wird auch die eigentliche Kettenaufgabe zum Vektor der gelösten Aufgaben hinzugefügt.

5.2.2 Aufgabenstellung - Einzelne Aufgaben - Kreativaufgaben

Dieses Modul besteht aus zwei Teilen: Der erste ist auf dem Server eingebunden, der zweite beim Client. Zuerst werde ich auf die Erfassung von Kreativaufgaben im Spieleeditor eingehen. Anschliessend auf die Bearbeitung und das Zurücksenden der Lösung. Die Korrektur und Bewertung folgt in einem späteren Modul.

5.2.2.1 Erfassung von Kreativaufgaben

Die folgende Abbildung zeigt wie der Spielleiter eine Kreativaufgabe im Spieleditor erfasst:

Abbildung 16: Erfassung einer Kreativaufgabe im Spieleditor

In der Tabelle werden die einzelnen Felder erläutert:

Feld	Beschreibung
Name of the Task?	Name und Identifikator der Kreativaufgabe. Dieser Name muss im ganzen Spiel einmalig sein und darf in keinem anderen Task verwendet werden.
Question?	Dieses Feld beinhaltet die eigentliche Aufgabe. Hier gibt der Spielleiter die Aufgabenstellung ein. Auch dieses Feld muss zwingend ausgefüllt werden, damit das System die Aufgabe akzeptiert.
Answer?	Die Lehrperson hat die Möglichkeit eine Musterlösung einzugeben. Diese wird ihr bei der Korrektur angezeigt und soll die Bewertung erleichtern.
Radius?	Definiert wie weit entfernt ein PDA sich maximal von der Position der Aufgabenstellung befinden darf, damit der Task noch gelöst werden kann. (Dies soll verhindern, dass ein Schlaumeier alle Aufgaben von der Mensa aus bei einem gemütlichen Kaffee lösen kann und sich nicht auf dem Campus bewegen muss.)
Correction Scheme?	Legt das Korrekturschema fest, welches bei dieser Kreativaufgabe verwendet werden soll. (Ausführlichere Informationen in den Kapiteln 5.2.5 und 5.2.6)
Group Work?	Definiert, ob diese Aufgabe alleine oder mit anderen Teams innerhalb der Gruppe gelöst werden soll. (Ausführlichere Informationen im Kapitel 5.2.3)

Tabelle 11: Erläuterung der Felder bei der Erfassung von Kreativaufgaben

Mit einem Klick auf den Button [OK] überprüft das System, ob alle Vorgaben eingehalten wurden und fügt die Aufgabe dem Spiel hinzu.

Die Kreativaufgabe ist eine Subklasse der Task-Klasse. Der statische Aufbau kann der nächsten Abbildung entnommen werden.

Abbildung 17: UML-Klassendiagramm: *CreativeTask.java*

5.2.2.2 Beantworten von Kreativaufgaben

Die nachfolgende Abbildung zeigt auf wie eine solche Kreativaufgabe auf dem PDA des Spielers dargestellt wird, nachdem der Spieler auf das Aufgaben-Symbol in der Karte geklickt hat.

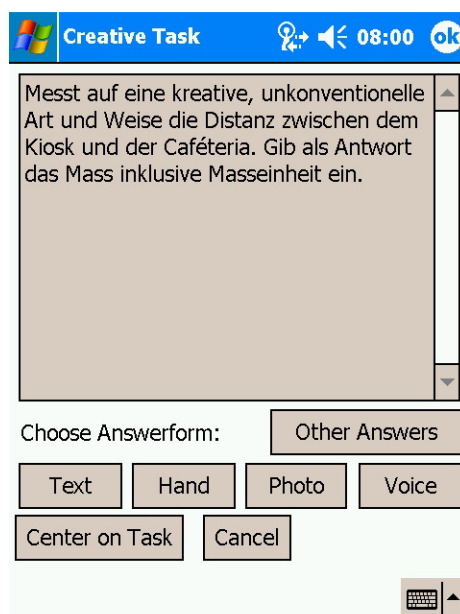


Abbildung 18: Kreativaufgabe auf dem PDA angezeigt

In der Tabelle werden die einzelnen Buttons genauer erläutert:

Button	Beschreibung
Other Answers	Zeigt die bereits abgegebenen Antworten der anderen Teams an. Eine ausführlichere Beschreibung folgt weiter unten mit dem Modul „Andere Antworten anzeigen“.
Text	Dieser Button ermöglicht es, dass der Spieler seine Antwort mit der Bildschirm-Tastatur des PDAs eingeben kann. Die Antwort wird mit einem String-Objekt gespeichert.
Hand	Mit einem Klick auf „Hand“ wird eine Handwriting-Annotation ermöglicht. Es wird eine Zeichenfläche aktiviert, auf welcher der Spieler seine Lösung mit unterschiedlichen Farben aufzeichnen kann. Die Handwriting-Annotation ist quasi ein digitales Blatt Papier mit Farbstiften.
Photo	Mit einem Klick auf „Photo“ wird eine Photo-Annotation ermöglicht. Hat der PDA eine eingebaute Fotokamera, wird diese aktiviert. Der Spieler fotografiert seine Lösung und überträgt sie an den Server.
Voice	Mit einem Klick auf „Voice“ wird eine Voice-Annotation ermöglicht. Das Team kann also eine Lösung dem Gerät diktieren und diese aufgenommene Stimme an den zentralen Rechner übertragen.
Center on Task	Dieser Button wurde von der Superklasse <i>NGQuestionDialog</i> geerbt. Mit einem Klick darauf wird wieder die Karte und das aktuelle Aufgabensymbol angezeigt.
Cancel	Bricht die Eingabe der Antwort ab und schliesst das Fenster.

Tabelle 12: Erläuterung der Buttons bei einer Kreativaufgabe

Die drei Annotationen sind an den Code angelehnt, welcher Christoph Wirz in seiner Diplomarbeit entwickelte. Damit sein Code aber in der ursprünglichen Form vorhanden blieb, duplizierte ich ihn zuerst und passte dann die Kopie entsprechend an.

5.2.2.3 *CreativeTaskResponse*

Damit die Antwort sinnvoll von Client und Server bearbeitet werden kann, wird ein neues Datenobjekt eingeführt: *CreativeTaskResponse*. Dieses Objekt beinhaltet alle Informationen, welche für die Bearbeitung, Anzeige und Korrektur der Spielerlösung notwendig sind. Für die Erfassung sind dies:

- Teamnamen: damit die Lösung dem Ersteller zugeordnet werden kann.
- Tasknamen: für die einfache Zuordnung der vorliegenden Antwort zur dazugehörenden Aufgabe.
- ID: besteht aus dem Hashcode von Teamnamen und Tasknamen. Damit kann die Antwort eindeutig aus allen erhaltenen Lösungen identifiziert werden.
- Text-Antwort: Speichert die Texteingabe des Spielers.
- Annotation: Hat der Spieler eine Annotation als Lösung erfasst, so wird diese in der Variable „Annotation“ gespeichert.

Komplett aufgeführt ist diese Klasse im folgenden UML-Klassendiagramm:



Abbildung 19: UML-Klassendiagramm: *CreativeTaskResponse.java*

Diese neu erstellte Klasse *CreativeTaskResponse.java* wird dann generiert, wenn der Spieler seine Lösung abschicken will und auf den „Senden“-Knopf gedrückt hat. Nach der Erstellung der Lösung überträgt der PDA die Antwort zum Server. Die Annotationen werden mit dem gleichen Prinzip versendet, wie dies Christoph Wirz in seiner Diplomarbeit (S. 34ff) beschrieben hat. Das blaue Aufgaben-Symbol der Kreativeaufgabe auf der Karte des Spielers wird nach dem Übermitteln entfernt und durch ein grünes i ersetzt. Die Funktionen des grünen i's werden im Kapitel 5.2.8.2 aufgeführt. Zusätzlich wurden folgende, für die Übermittlung benötigte Ereigniskennungen in die Klasse *GameEvents* hinzugefügt:

Kennung	Verwendung
NEW_CREATIVETASK_EVENT	Wenn der Spieler eine neue Antwort versendet, wird diese Kennung verwendet, damit der Server klar erkennt, welchen Inhalt die empfangene Nachricht hat und dass eine neue Kreativaufgaben-Antwort vorliegt.
NEW_CREATIVETASK_ID_EVENT	Der Spieler erstellt eine Annotation in der Lösung. Damit diese vom Client zum Server übertragen werden kann, ist es notwendig, dass der Server einen eindeutigen Identifikator dem Client sendet. Dieser klassifiziert die Datenpakete, um beim Zusammensetzen der einzelnen Annotationen keine Lücken und Überschneidungen im Datenstrom entstehen zu lassen.
NEW_CREATIVETASK_DATA_EVENT	Sendet nun der PDA ein Datenpaket mit Teilen der zu versendenden Annotation, wird die Netzwerknachricht mit der links stehenden Kennung markiert.
NEW_CREATIVETASK_DATA_ACK_EVENT	Wenn der Server das empfangene Datenpaket zur Spielerantwort hinzufügen konnte, sendet er eine Bestätigungsmeldung zurück an den Client. Eine vom Client gesteuerte Übertragungswiederholung wird dann hinfällig.

Tabelle 13: Ereigniskennungen für die Übertragung einer Spielerlösung

Abbildung 20 zeigt systematisch die Nachrichten auf, welche zwischen Server und Client ausgetauscht werden, damit der Server die Spielerantworten ohne Übertragungsfehler zusammensetzen kann.

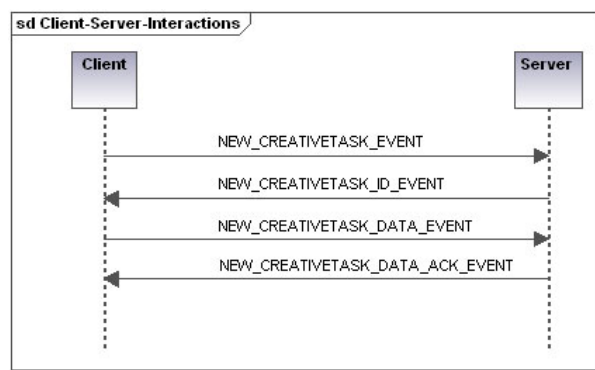


Abbildung 20: Sequenzdiagramm: Übertragen einer Spielerantwort (Übersicht)

5.2.2.4 Erfassen und versenden einer Spielerantwort

Jetzt nachdem die Grundlagen erläutert wurden und der Gesamtzusammenhang erklärt ist, möchte ich noch etwas detaillierter auf das Lösen und Versenden einer Spielerantwort eingehen. In Abbildung 21 ist grafisch dargestellt, wie das Erstellen und Versenden der Antwort clientseitig funktioniert und wann auch die in der vorhergehenden Abbildung aufgeführten Übertragungsereignisse verwendet werden. Es wird beispielhaft das Lösen einer Kreativaufgabe mit Photo-Annotation aufgezeigt.

Klickt der Spieler auf der Spielkarte ein Aufgabensymbol einer Kreativaufgabe an, wird ein neues Dialogfenster *NGCreativeTaskDialog(..)* geöffnet (siehe dazu auch Abbildung 18). Dieses Dialogfenster erstellt direkt ein neues *CreativeTaskResponse(..)*-Objekt. Der Spieler will die Frage mit einem Foto beantworten. Er klickt daher auf den Button [Photo]. *handlePhoto()* initialisiert nun ein neues Dialogfenster *NGPhotoCreativeTaskDialog(..)*. Mit der Methode *startRecording()* erzeugt der Spieler dann das Foto. Hat er das Bild geschossen, zeigt ihm die Methode *loadImage()* seine Aufnahme an. Hier hat er die Möglichkeit das Foto zu verwerfen und ein neues zu schießen oder er kann es bestätigen und mit einem Klick auf „Submit“ an

den Server versenden. Wählt er „Submit“ aus, wird einfach das Dialogfenster geschlossen. Die vorhin gewählte Methode *handlePhoto()* liest nun das Foto (die Annotation) vom Objekt *NGPhotoCreativeTaskDialog(..)* aus und speichert es im initialisierten Objekt *CreativeTaskResponse* ab. Auch der Name der Aufgabe wird nun gespeichert, damit die Antwort später wieder gefunden werden kann. Nachdem alle diese Aktionen durchgeführt wurden, wird die Methode *sendResponse()* aufgerufen. Diese Methode übergibt das Antwortobjekt der *CreativeTaskResponseLogic*, welche den Versand der Antwort an den Server regelt. Zuerst wird die Annotation im *GameStore* zwischengespeichert. Die eigentliche Annotation wartet so auf den Versand an den Server bis eine eindeutige Identifikationsnummer von diesem erhalten wurde. Die restlichen Informationen der *CreativeTaskResponse* werden über den *ClientNetworkHandler* mit der Methode *sendMessageByTCP(..)* an den Server versandt. Um dem Server zu markieren, dass diese Nachricht eine neue Spielerantwort ist, wird sie mit der Kennung *GameEvents.NEW_CREATIVETASK_EVENT* deklariert. Der Server verarbeitet dann diese Nachricht (im nachfolgenden Kapitel beschrieben) und sendet dem Client eine ID für das Versenden der Annotation zurück. Die ID ist mit der Kennung *GameEvents.NEW_CREATIVETASK_ID_EVENT* markiert. Die Klasse *GameStore* fügt sie dann der wartenden Annotation hinzu.

Gleichzeitig mit dem Übergeben der Antwort an die *CreativeTaskResponseLogic* wird ein Dialogfenster der Klasse *NGCreativeTaskTransferDialog* geöffnet. Dieses zeigt dem Spieler jeweils den aktuellen Stand der Übertragung seiner Antwort an. Solange die Annotation noch auf die Übermittlung wartet, wird die Meldung „wait for server...“ angezeigt. Im Hintergrund wird in einer While-Schleife immer wieder abgefragt, ob die Annotation mittlerweile die ID vom Server zugewiesen bekam.

Wurde die ID der Annotation hinzugefügt, wird in einer weiteren While-Schleife die Annotation mit der Methode *sendNextDataBlock(BLOBAnnotation)* der *CreativeTaskResponseLogic* in neue *BLOBAnnotationDataItems* unterteilt und einzeln versendet. Anschliessend werden diese *BLOBAnnotationDataItems* mit der bereits bekannten Methode *sendMessageByTCP(..)* und der Kennung *GameEvents.NEW_CREATIVETASK_DATA_EVENT* an den Server geschickt. Bis der Server den Empfang quittiert hat, werden die einzelnen DataItems im *GameStore* zwischengespeichert. Der Client wartet nun. Trifft dann die Bestätigung ein (*GameEvents.NEW_CREATIVETASK_DATA_ACK_EVENT*), meldet die *CreativeTaskResponseLogic* dem *NGCreativeTaskTransferDialog* die erfolgreiche Übermittlung. Gibt es Übertragungsprobleme, wird die Übertragungswiederholung zeitgesteuert vom *NGCreativeTaskTransferDialog* in Gang gesetzt.

Nachdem alle Datenblöcke übertragen und bestätigt wurden, wird der Status des *NGCreativeTaskTransferDialog* auf TRUE gesetzt. Die Klasse *NGCreativeTaskDialog*, welche den ganzen Prozess gestartet und unter Kontrolle hat, speichert nun die gelöste Aufgabe ab und „legt“ sie zu den anderen gelösten Aufgaben. Nun werden alle Statistiken nachgeführt. Schliesslich wird auch das erste Fenster geschlossen und die Karte mit den einzelnen noch ungelösten Aufgaben wieder angezeigt.

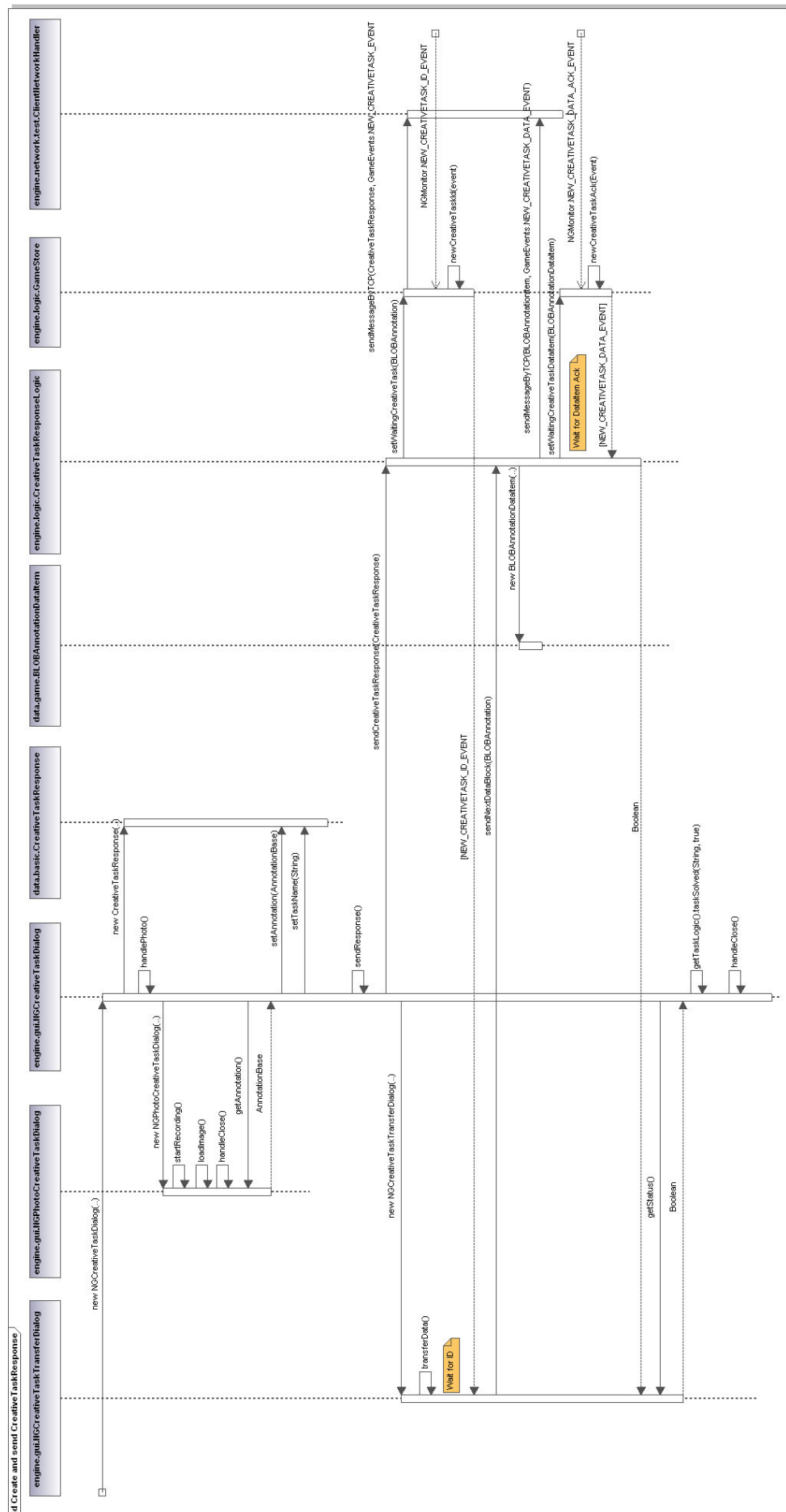


Abbildung 21: Sequenzdiagramm: Übertragung einer Spielerantwort (Client)

5.2.2.5 Verarbeitung einer Antwort

Trifft eine in Tabelle 13 aufgeführte Nachricht beim Server ein, wird diese durch den *CreativeTaskHandler* abgefangen und bearbeitet. Die Klasse des *CreativeTaskHandlers* ist nebst den Funktionen, die er vom *TaskHandler* erbt, wie folgt aufgebaut:

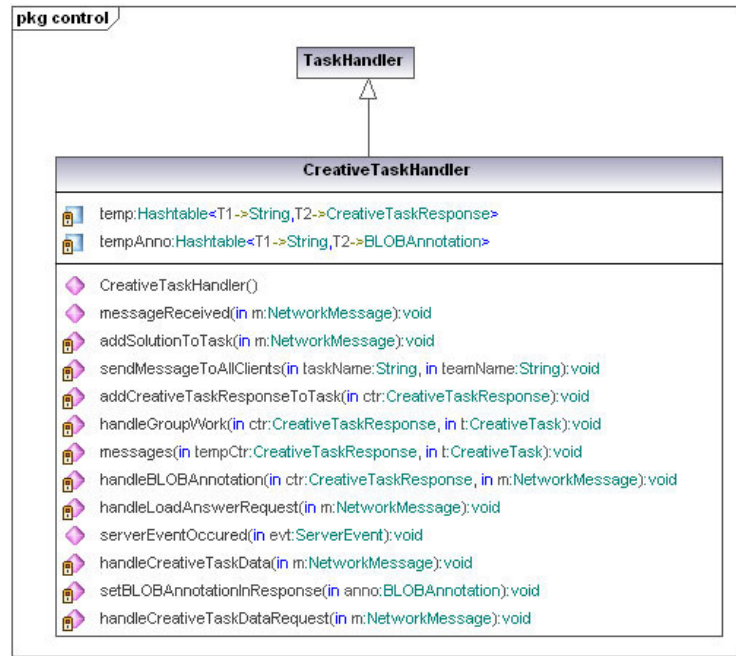


Abbildung 22: UML-Klassendiagramm *CreativeTaskHandler.java*

Die Funktionen der einzelnen Methoden wird auf den nächsten paar Seiten genauer erläutert. Abbildung 23 zeigt die notwendigen Schritte auf, welche beim Empfang einer Spielerlösung abgearbeitet werden müssen.

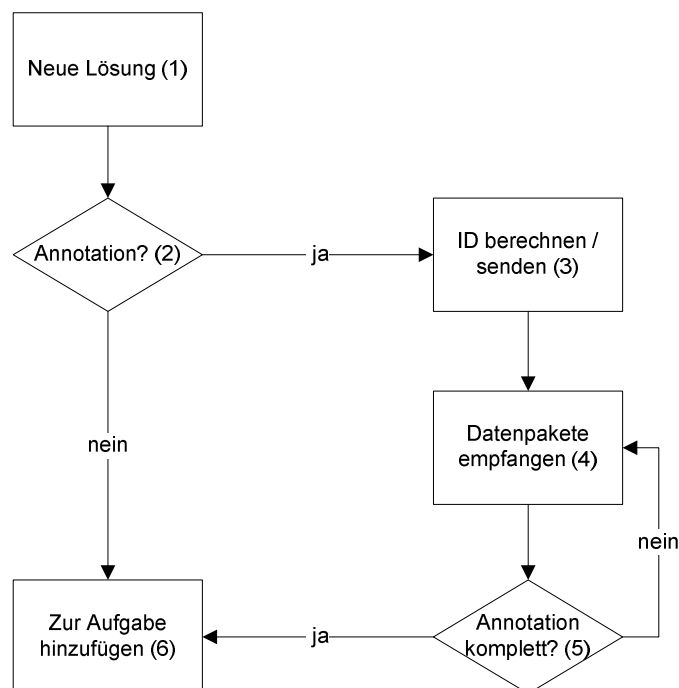


Abbildung 23: Erhalt einer *CreativeTaskResponse*: Verarbeitung im *CreativeTaskHandler*

Die einzelnen Arbeitsschritte werden nun beschrieben:

1. Die neue Spielerantwort trifft ein. Diese wird sofort auf dem Server zwischengespeichert.
2. Der *CreativeTaskHandler* überprüft nun, ob die Spielerlösung eine Annotation oder eine Textantwort beinhaltet. Hat die Lösung nur eine Textantwort, kann direkt zu Punkt 6 übergegangen werden.
3. Eine Annotation ist vorhanden. Der Server berechnet daher eine einmalige ID. Die ID ist der Hashcode der Konkatenation von Team- und Tasknamen. Diese ID ist aufgrund der Tatsache einmalig, dass jedes Team jede Aufgabe nur einmal lösen darf. Mit dieser ID wird einerseits eine temporäre Annotation auf dem Server gespeichert und andererseits wird sie dem PDA gesandt, von welchem die Lösung stammt.
4. Der PDA versendet nun mit dieser ID die Datenblöcke der Annotation. Der Server kann die Daten durch die ID identifizieren und weist sie direkt der temporären Annotation zu. Damit der Client weiss, dass die Zentrale die Lösung erhalten hat, sendet der Server eine Empfangsbestätigung zurück.
5. Anschliessend überprüft der *CreativeTaskHandler* ob die Annotation komplett übertragen wurde. Wenn dies nicht der Fall ist, empfängt er weitere Daten vom Client (zurück zu 4). Konnte die Annotation mit dem aktuellen Datenpaket vervollständigt werden, kann nun die komplette Lösung der Aufgabenantwort zugewiesen werden.
6. Um ein einfaches Verwalten der Spielerlösungen zu ermöglichen, beinhaltet die Klasse *CreativeTask* eine Hashtabelle in welcher alle Spielerlösungen einer Aufgabe gespeichert werden. Als Schlüssel zum Wiederaufrufen der Lösung wird der Teamname verwendet. Der *TaskHandler* speichert schliesslich mit seiner eigenen Ablaufroutine den Task als gelöst ab. So wird versichert, dass das System immer konsistent bleibt.

5.2.3 Lösen der Aufgabe - Gruppenaufgaben

Als nächstes Modul wird das Lösen der Aufgaben in Gruppen beschrieben.

Will der Spielleiter, dass eine Kreativaufgabe als Gruppenaufgabe gelöst wird, kann er beim Erfassen der Kreativaufgabe einfach das Kästchen „Group Work“ anwählen (siehe dazu Abbildung 16).

Ruft nun der Spieler diesen Task auf, wird ihm unten rechts ein Button [GroupWork] angezeigt. Klickt er auf diesen, wird das Fenster geöffnet, welches unten abgebildet ist. Der Spieler muss also vor der Antwortfassung wählen mit welchen anderen Teams seiner Gruppe er diese Aufgabe lösen will. Er kann ein, zwei, alle oder kein Team anwählen. Wählt er kein Team an, geht das System davon aus, dass er die Aufgabe alleine löst. Mit dem Klick auf [save] speichert der Rechner die ausgewählten Teamnamen in einem Vektor. Anschliessend wird das bekannte Aufgabenfenster von Abbildung 18 dargestellt. Der Ablauf nach der Auswahl der beteiligten Teams ist derselbe wie in 5.2.2.2 beschrieben ist.

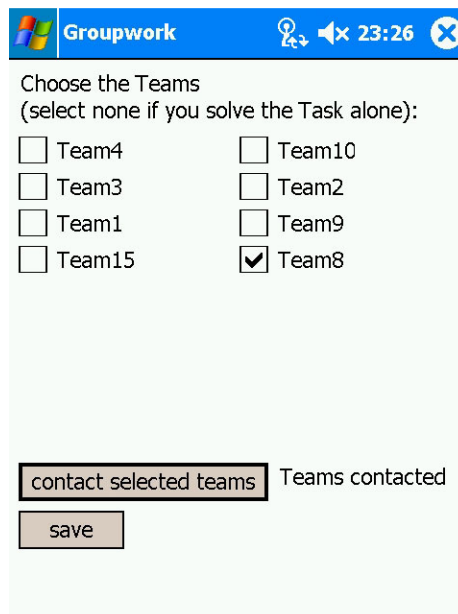


Abbildung 24: Auswahl der beteiligten Teams

5.2.3.1 Organisieren der Teams für eine Gruppenarbeit

Klickt der Spieler innerhalb der Ansicht von Abbildung 24 den Button [contact selected teams] an, nachdem er die gewünschten Teams ausgewählt hat, werden alle selektierten Teams informiert, dass ein anderes Team eine Gruppenaufgabe gemeinsam lösen möchte. Auf dem Bildschirm der kontaktierten Teams erscheint nun ein Dialogfenster wie es in der folgenden Abbildung dargestellt ist.

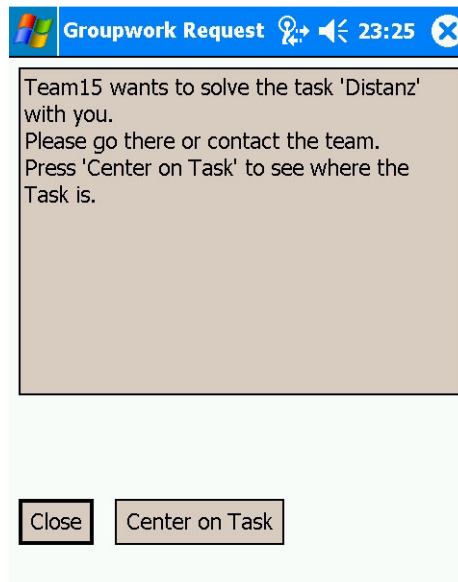


Abbildung 25: Groupwork-Request: Anfrage eines anderen Teams zum gemeinsamen Lösen einer Gruppenaufgabe

In der Textbox steht drin welches Team welche Aufgabe gemeinsam lösen möchte. Der Spieler wird nun aufgefordert, zu diesem Task hinzugehen oder mit dem anfragenden Team Kontakt aufzunehmen. Der Spieler hat zwei Möglichkeiten das Fenster zu schliessen: Klickt er nur auf den Button [Close] wird das Fenster ohne Veränderung an der Karte geschlossen. Wählt er hingegen den zweiten Button [Center on Task] an, wird das Dialogfenster auch beendet, die Karte wird aber auf die zu lösende Aufgabe ausgerichtet.

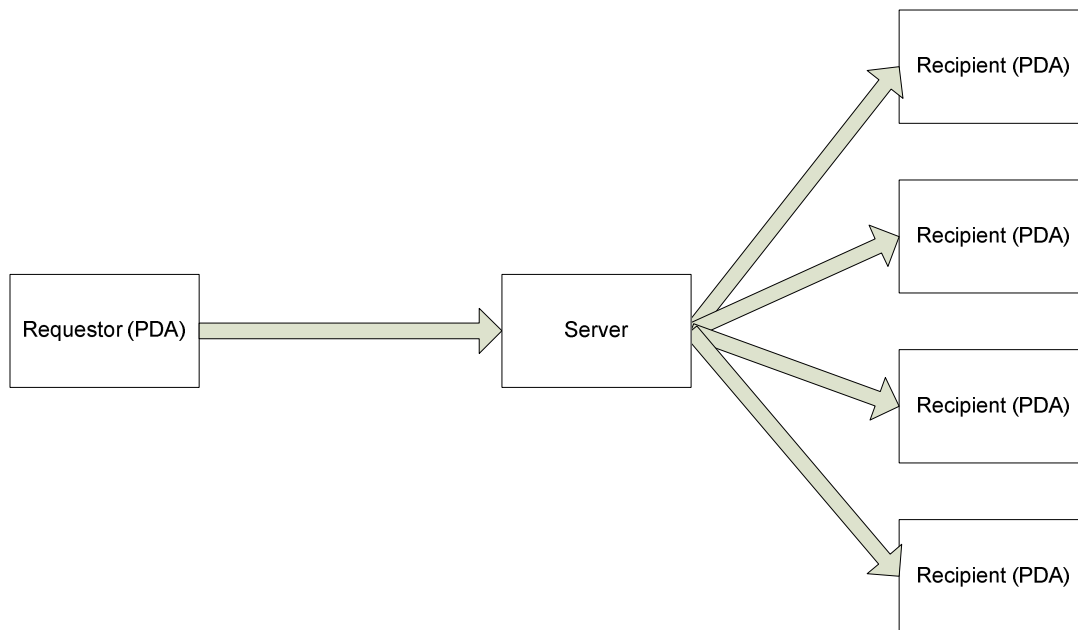


Abbildung 26: Übersicht über das Versenden von Groupwork-Requests

Der oben aufgeführte schematische Ablauf zeigt, dass der sendende Client nur eine Nachricht an den Server sendet und der Server dann die Nachricht vervielfältigt und an die einzelnen ausgewählten Teams weiterleitet. Es steht die Absicht dahinter, eine möglichst kleine Anzahl Nachrichten versenden zu müssen um die Netzwerklast so tief wie möglich zu halten. Dies ist mit der gewählten Lösung der Fall.

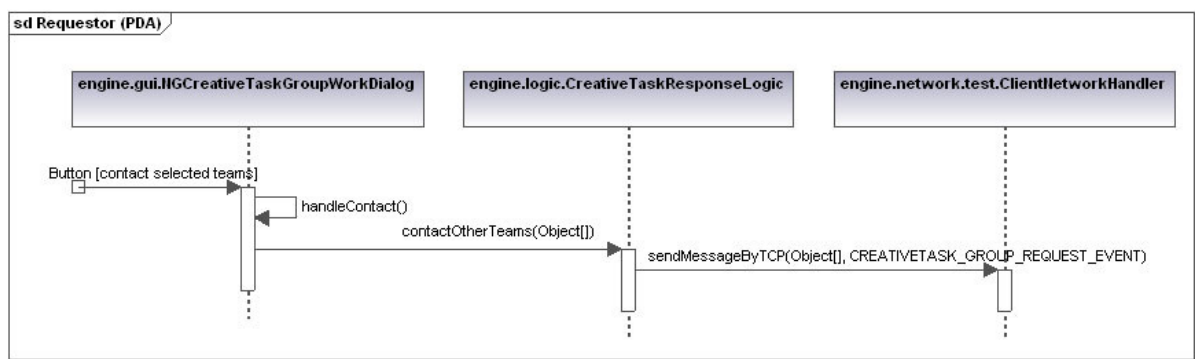


Abbildung 27: Sequenzdiagramm: Versenden von Groupwork-Request (Sicht Requestor)

Hat der Spieler die Teams angewählt und auf den Button [contact selected teams] gedrückt, erzeugt die Methode *handleContact()* ein Objekt-Array. An der ersten Position wird der Taskname gespeichert. An der zweiten Stelle der Name des Requestor-Teams und in der dritten ein Vektor, welcher die Namen der ausgewählten Teams enthält. Dieses Objekt-Array wird durch die Klasse *CreativeTaskResponseLogic* über den *ClientNetworkHandler* an den Server versendet. Die Netzwerknachricht wird mit dem neuen Event *CREATIVETASK_GROUP_REQUEST_EVENT* gekennzeichnet.

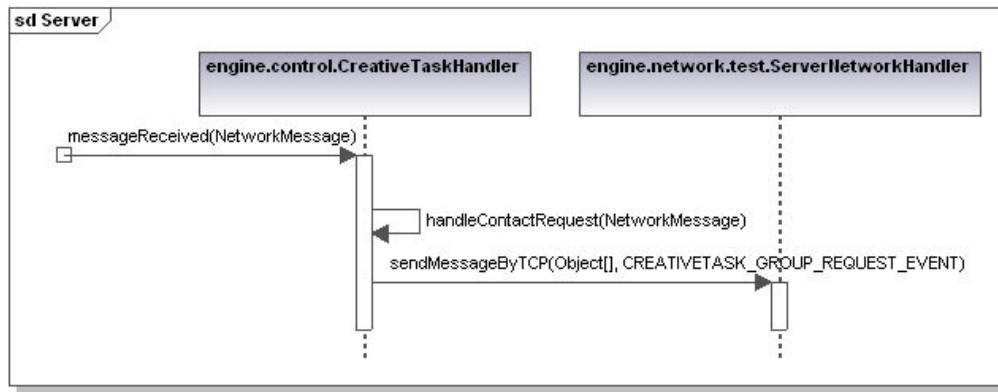


Abbildung 28: Sequenzdiagramm: Weiterleiten von Groupwork-Requests (Sicht Server)

Serverseitig wird die vorhin versendete Netzwerknachricht von der Klasse *CreativeTaskHandler* abgefangen. Die Methode *handleContactRequest(..)* verarbeitet die erhaltene Nachricht. Zuerst wird der Vektor mit allen Teams zwischengespeichert. Mit einer for-Schleife versendet der Server den Aufgaben- und Requestor-Namen an die Teams, deren Namen sich im Vektor befinden. Es wird wieder dieselbe Netzwerknachricht-Kennung wie vorhin verwendet.

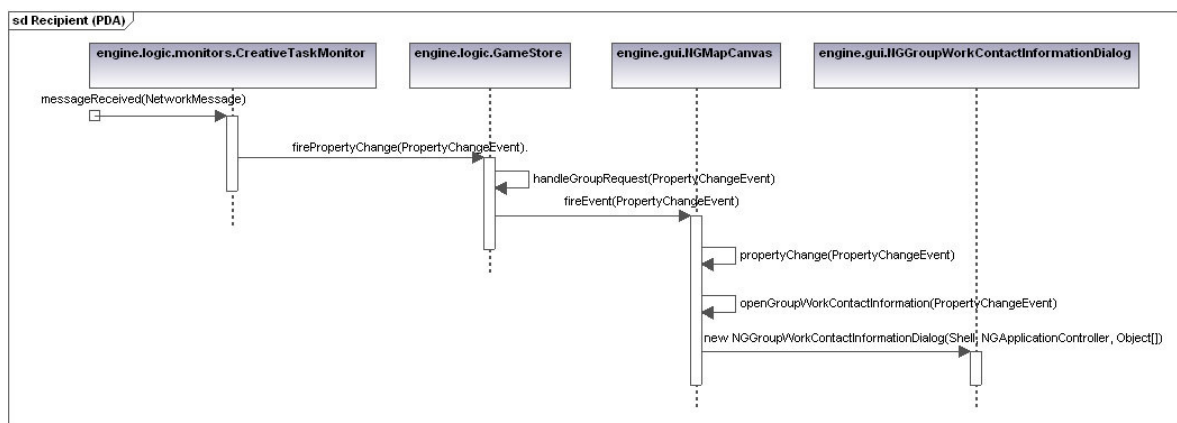


Abbildung 29: Sequenzdiagramm: Empfangen einer Groupwork-Request (Sicht Recipient)

Die Nachricht wird beim Client standardmässig vom *CreativeTaskMonitor* bearbeitet und mit der Methode *firePropertyChange(..)* im System verteilt. Die Klasse *GameStore* fängt diese Nachricht ab und wirft sie mit der Methode *fireEvent(..)* so auf, dass *NGMapCanvas* diesen Event erfassen und das in Abbildung 25 dargestellte Dialogfenster durch die Methode *openGroupWorkContactInformation(..)* initialisieren und auf dem PDA einblenden kann.

5.2.3.2 Serverseitiger Empfang einer Gruppenarbeit

Nun geht es um die Bearbeitung einer vom Client empfangenen Gruppenantwort. Um eine Gruppenarbeit adäquat zu verwalten, werden die Funktionen des *CreativeTaskHandlers* erweitert. Es soll für jedes an der Lösung beteiligte Team eine Kopie der Antwort erstellt werden. So wird dem Computer vorgegaukelt, dass diese Lösung von diesem einzelnen Team ist. Dies ist notwendig, damit ein Team während dem Spiel problemlos den PDA neustarten kann. Die folgende Abbildung gibt eine Übersicht über die Behandlung von Gruppenantworten durch den *CreativeTaskHandler*.

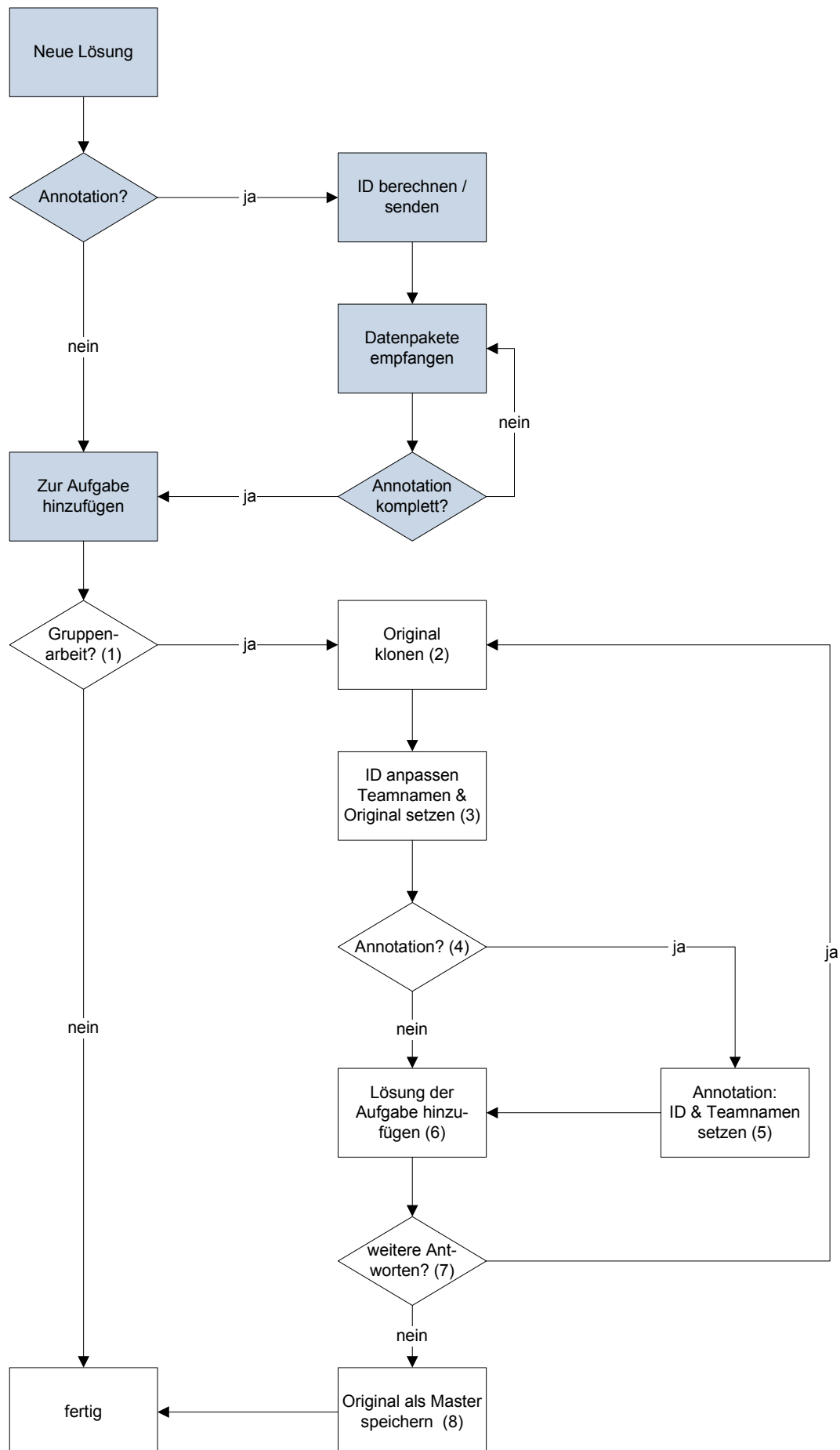


Abbildung 30: Erweiterung des CreativeTaskHandlers um die Gruppenarbeit

1. Im ersten Schritt wird überprüft, ob der empfangene Task eine Gruppenaufgabe ist.
2. Für das erste Team im Teamnamen-Vektor wird nun die ursprünglich erfasste Antwort geklont (dupliziert). Diese Klon-Funktion zwingt den Rechner dazu, eine neue Antwort in den Speicher zu schreiben und nicht einfach eine neue Referenz auf ein schon gespeichertes Element zu erzeugen.
3. Nun werden alle „teamspezifischen“ Variablen angepasst: Die ID wird neu berechnet, der Teamnamen des Senders wird durch den aktuellen Teamnamen ersetzt. Zusätzlich wird in einer spezifischen Variable der Name des Originals gespeichert. Diese Variable wird dann bei der Bewertung der Lösung eine wichtige Rolle spielen.
4. Nun wird überprüft, ob das Original eine Annotation hat.
5. Da das Original eine Annotation hat, ist es notwendig, dass auch die geklonte Annotation angepasst wird. So sollen allfällige Probleme schon vor Auftreten beseitigt werden. Es werden daher der Annotation eine dem Antwortobjekt angepasste ID und den richtigen Teamnamen gegeben.
6. Während die Originalantwort schon früher der Aufgabe zugewiesen wurde, wird jetzt auch der angepasste Klon dem Task hinzugefügt.
7. Es wird überprüft, ob noch weitere Teams an der Antwort beteiligt sind. Ist dies der Fall, wird diese Prozedur auch für diese Teams durchgeführt. So haben alle Teams eine „eigene“ Antwort.
8. Ganz am Ende wird die Originalantwort speziell markiert, damit der Server erkennen kann, dass hier die ursprüngliche Lösung vorliegt. Dies kann erst am Ende der Prozedur gemacht werden, da sonst diese Einstellung auch mitgeklont würde und bei allen duplizierten Antworten zurückgesetzt werden müsste, was schliesslich einen höheren Aufwand mit sich bringen würde.

Wenn nun eine solche Aufgabe auf dem Server gespeichert wurde, ist es notwendig, dass auch die Gebäudekarte der Teams, welche an der Lösung beteiligt sind, aber nicht abschickten, angepasst wird. Es können zwei Fälle unterschieden werden:

- Wenn ein Task von einem anderen Team gelöst wurde, verschwindet die Aufgabe auf der Karte und wird durch ein grünes i ersetzt. So ist es keinem Team möglich, eine zweite Lösung abzugeben. Es werden zusätzlich immer Nachrichten („Team A hat die Aufgabe m gelöst“) an alle verbundenen PDAs gesendet und von diesen abgespeichert. Dadurch kann sichergestellt werden, dass alle nachfolgenden Teams die Aufgabe nur noch mit Teams lösen, welche diese Aufgabe noch nicht absolvierten. Ist eine Aufgabe zusätzlich noch Teil einer Kettenaufgabe, wird automatisch der nachfolgende Task angezeigt. Dies muss separat geregelt werden und wird von der Klasse *TaskLogic* übernommen.
- Ein Team hat für die komplette Gruppe eine Antwort gesendet. Nun möchte wenige Sekunden später ein anderes Team von dieser Gruppe eine Lösung senden. Auch diese Antwort gelangt zum Server. Der *CreativeTaskHandler* filtert diese zweite Antwort aus und verwirft sie. Eine Gruppe kann keine zweite Lösung einreichen.

5.2.4 Korrektur / Bewertung - Spielleiter

In den kommenden Kapiteln wird die dritte Phase des vorgestellten Lernprozesses beschrieben: Korrektur und Bewertung.

Damit der Spielleiter überhaupt sieht, welche Aufgaben er korrigieren muss, wurde im Spieleditor ein weiteres Register „Creative Tasks“ hinzugefügt:

Team	Received	Corrected	Rating	Groupwork
team1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1 (10)	Team Solution: team2
team3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2 (0)	

Buttons: correct

Abbildung 31: Übersicht über die empfangenen Antworten

Dieses Register ist wie folgt aufgebaut: Jede einzelne Kreativaufgabe hat ein eigenes Register (1). In diesem Register werden alle Lösungen zu dieser Aufgabe aufgeführt. Die Spielerantworten sind in einer Tabelle angeordnet: Team (2) zeigt auf, wer diese Aufgabe abgeschickt hat. Bei einer Gruppenarbeit wird hier der Teamname des Senders der Lösung angegeben. Die Spalte „Received“ (3) zeigt den Status der Antwort an: Ist ein Häkchen gesetzt, hat dieses Team die Aufgabe gelöst und die Antwort wurde auf dem Server gespeichert. Bei Corrected (4) steht das Häkchen hingegen nur, wenn der Spielleiter die Aufgabe schon korrigiert hat. Ist dort kein Häkchen, muss er die Lösung des Teams noch korrigieren. Die Rating-Spalte (5) gibt an, wie viele Punkte der Spielleiter für diese Lösung gegeben hat beziehungsweise welchen Rang das Team erzielte. Im vorliegenden Fall hat team1 den ersten Rang und ist somit besser bewertet als team3. Die Spalte „Groupwork“ zählt die Teams auf, welche bei dieser Lösung mitwirkten. Ein Spielleiter kann eine Antwort korrigieren, in dem er das gewünschte Team auswählt und auf den „correct“-Button (7) drückt oder auf dem Teamnamen doppelklickt. Dann wird je nach ausgewähltem Bewertungsschema ein Fenster für die Rating Score- bzw. Nominalskala-Bewertung geöffnet. Diese beiden Korrekturfenster werden in den folgenden zwei Abschnitten vorgestellt.

5.2.5 Korrektur / Bewertung - Rating Score

Die Korrektur nach dem Rating Score-Schema wird in folgendem Dialogfenster vorgenommen:

Correction - team3

Question: „Erweitert“ diese Skulptur: stellt euch passend zu dieser Skulptur hin und lasst euch von einem Passanten fotografieren. Lasst euch von den anderen Antworten inspirieren.

Client-Answer:

Solution: (keine Musterlösung verfügbar)

Comment:

Points (max: 10):

Buttons: Cancel, Save

Abbildung 32: Rating Score-Dialogfenster

Auf der linken Seite wird oben die Aufgabenstellung angezeigt und unten die Musterlösung, welche vom Spielleiter eingegeben wurde. Dies soll die Korrektur vereinfachen, da der Korrigierende alle notwendigen Angaben in einem Fenster dargestellt erhält. In der Mitte wird die Spielerlösung eingeblendet. Bevor das Fenster angezeigt wird, entscheidet der Rechner je nach vorliegender Antwort, ob er ein Textfeld, ein Feld, in dem Bilder dargestellt werden können, oder ein Button für das Abspielen von Voice-Antworten darstellen muss. Auf der rechten Seite hat der Spielleiter die Möglichkeit seine Bewertung zu kommentieren. Dieser Kommentar wird nicht an den Client gesendet. Dies ist bewusst so gehalten um den Spieler während des Spiels nicht abzulenken. Der Kommentar wird mit dem HTML-Export (Kapitel 5.2.8.3) den Spielern zugänglich gemacht. Unten rechts kann der Spielleiter die Punkte eingeben. In der Klammer steht die maximal mögliche Punktzahl. Das System prüft automatisch die Einhaltung dieser Grenze. Der Maximalwert kann vor dem Spielstart in den Regeln erfasst werden. Standardmässig ist der Wert 10 eingestellt. Damit der Spielleiter immer sieht, welches Team er aktuell bewertet, wird der Teamname oben in der Titelleiste eingeblendet.

Mit einem Klick auf [save] wird die Punktzahl direkt im Antwortobjekt *CreativeTaskResponse* gespeichert und den Status der Antwort auf „corrected“ gesetzt. Zusätzlich werden die vergebenen Punkte auch noch im *TeamStatus* gespeichert. Nachdem alle diese Aktionen durchgeführt wurden, berechnet der Server die Rangliste neu und versendet sie den einzelnen Teams.

5.2.6 Korrektur / Bewertung - Nominalskala

Eine Kreativaufgabe wird in folgendem Fenster nach dem Nominalskala-Schema bewertet:

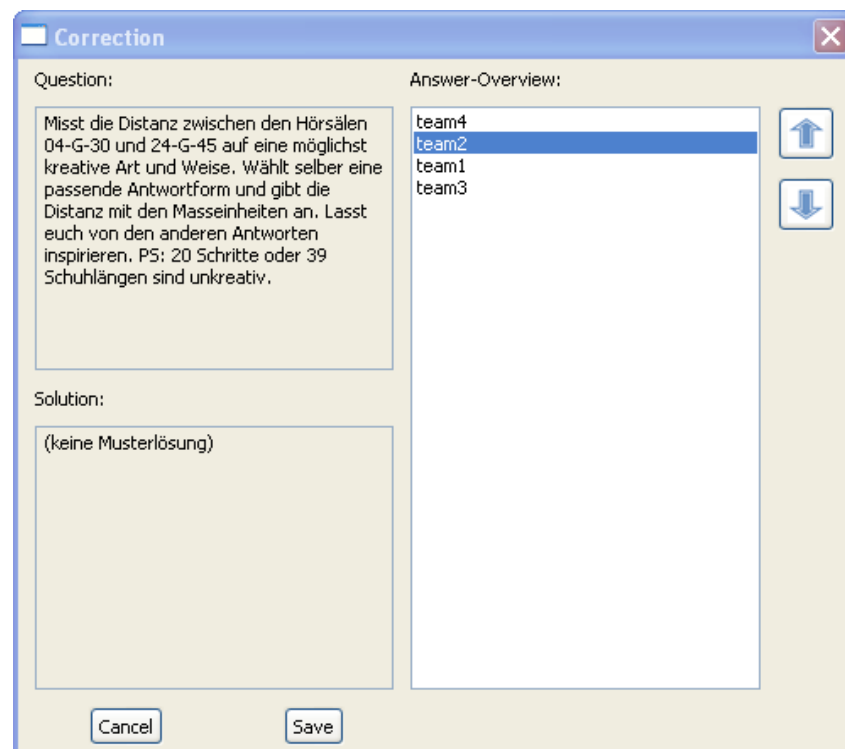


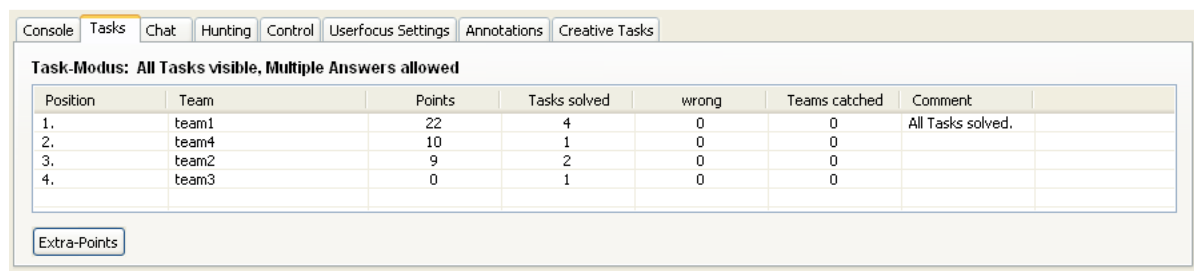
Abbildung 33: Nominalskala-Dialogfenster

Wie schon beim Rating Score-Dialogfenster (Abbildung 32) sind auf der linken Seite die Aufgabenstellung und die Musterlösung aufgeführt. In der Mitte sind alle Teams aufgelistet, deren Antworten zu korrigieren sind. Mit einem Doppelklick auf den Teamnamen wird die Lösung in einem neuen Dialogfenster angezeigt. Für den Spielleiter gilt es nun, die Teamnamen so zu ordnen, dass das beste Team die Rangliste anführt und das schlechteste zuunterst platziert wird. Im oben gezeigten Beispiel bedeutet dies in etwa:

Lösung von team4 ist besser als die Lösung von team2; Lösung von team2 ist besser als die Lösung von team1; etc. Mit den Auf- und Ab-Pfeilen wird die Position der Teams verändert. Klickt der Spielleiter auf den Button [save] werden die Reihenfolge gespeichert und die Punkte berechnet. Die beste Lösung erhält die maximal mögliche Punktzahl, während die schlechteste keine erhält. Alle anderen Aufgaben erhalten ihre Punkte linear abgestuft entsprechend der Rangierung. Jede Antwort hat eine Variable, in welcher der Server den aktuellen Rang speichert. Beim Speichern wird darauf geachtet, dass jede Rangierung nur einmal verwendet wird. Es gibt eine Ausnahme: Liegt eine Gruppenantwort vor erhalten alle beteiligten Teams den gleichen Rang und damit die gleiche Anzahl Punkte.

5.2.7 Feedback - Rangliste

Die verteilten Punkte werden in einer Rangliste aufgeführt. Dazu wurde die bereits programmierte Übersicht „Tasks“ im Spieleditor angepasst und übersichtlicher gestaltet. Der Spielleiter hat so auf eine einfache Art die Übersicht über die verschiedenen Teams.



The screenshot shows a software interface with a 'Tasks' tab selected. Below the tab, there is a header 'Task-Modus: All Tasks visible, Multiple Answers allowed'. A table displays the following data:

Position	Team	Points	Tasks solved	wrong	Teams caught	Comment
1.	team1	22	4	0	0	All Tasks solved.
2.	team4	10	1	0	0	
3.	team2	9	2	0	0	
4.	team3	0	1	0	0	

Below the table is a button labeled 'Extra-Points'.

Abbildung 34: Übersicht über die verschiedenen Teams

Die Klasse *GameStatus* speichert die Teamnamen und dazugehörigen Punkte in einem sortierten Vektor ab. Um die in Abbildung 34 dargestellte Tabelle zu erzeugen, wird in einer for-Schleife die Rangliste vom *GameStatus* abgearbeitet. Informationen wie viele Aufgaben ein Team schon richtig und falsch löste, werden direkt vom *TeamStatus* des entsprechenden Teams abgerufen. Die Spalte „Comment“ erhält den Text „All Tasks solved.“ wenn das Team alle im System vorhandenen Aufgaben erfolgreich löste. Die zweitletzte Spalte „Teams caught“ gibt an, wie oft der Spieler andere Teams fangen konnte.

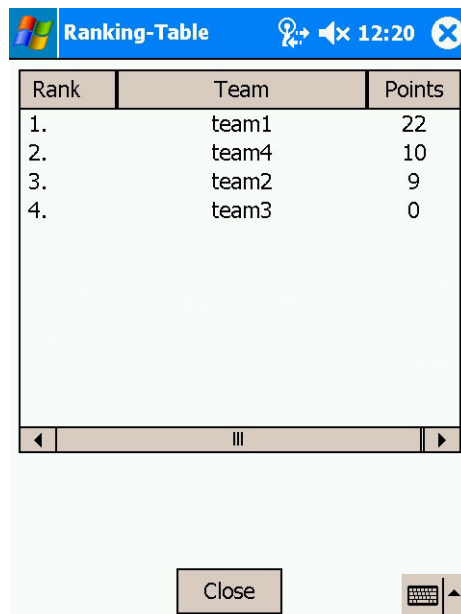
Die Punkte für „Nicht-Kreativaufgaben“ werden anhand des folgenden Schemas verteilt:

- Eine richtige Antwort gibt immer die maximale Punktzahl.
- Wurde eine Aufgabe zuerst falsch und anschliessend richtig gelöst, werden Punkte abgezogen: für jede falsche Antwort einen bestimmten, in den Regeln festgelegter Wert. Für die richtige Lösung gibt es aber wieder die volle Punktzahl. Das Punktetotal der Aufgabe errechnet sich aus der Summe von allen falschen Antworten multipliziert mit der Anzahl Abzugspunkte pro Fehler und der maximalen Punktzahl für die richtige Antwort. Eine Aufgabe kann als Minimum aber nicht weniger als 0 Punkte erhalten.
- Ein Team kann sich zusätzliche Punkte verdienen in dem es andere Teams jagt und fängt. Für jedes gefangene Team erhält der Fänger den in den Regeln vordefinierten Wert.
- Der Spielleiter kann Extrapunkte für ausserordentliche Leistungen verteilen. Dies ist genauer im Kapitel 5.2.8.1 beschrieben.

5.2.7.1 Rangliste auf dem PDA

Damit der Spieler auch während dem Spiel sieht, wie gut er und die anderen Teams momentan sind, hat er die Möglichkeit jederzeit die aktuelle Tabelle abzurufen. Da die Rangliste vom *GameStatus* immer selbstständig in einem Push-Modus an die verbundenen Geräte gesendet wird, kann der Spieler ohne Verzöge-

rung seine aktuelle Position begutachten. Mit einem Klick auf den „Goldenen Pokal“ in der Symbolleiste wird die unten stehende Ansicht auf dem PDA eingeblendet.



Rank	Team	Points
1.	team1	22
2.	team4	10
3.	team2	9
4.	team3	0

Abbildung 35: Anzeige der Rangliste auf dem Client

5.2.8 Zusätzliche Funktionen

5.2.8.1 Extrapunkte

Der Spielleiter hat die Möglichkeit nach seinem Erachten zusätzliche Punkte zu verteilen. Dazu klickt er auf den Button [Extra-Points] in der Registerkarte „Tasks“ (siehe Abbildung 34). Anschliessend wird das folgende Dialog-Fenster angezeigt:

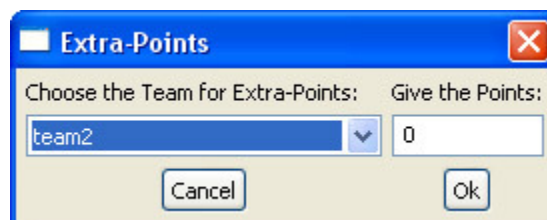


Abbildung 36: Extrapunkte - Dialogfenster

Auf der linken Seite wählt der Spielleiter das Team aus und vergibt dann auf der rechten Seite die Punkte. Wählt er anschliessend ein anderes Team aus, werden die neuen Punkte des vorhin selektierten Teams mit der Methode *setPointText()* im *TeamStatus* gespeichert. Damit dies möglich ist, wurde die Klasse *TeamStatus* um die Integer-Variable *extraPoints* erweitert. Hat der Spielleiter nun alle Extrapunkte verteilt, bestätigt er die Eingaben mit einem Klick auf [OK]. Der letzte Eintrag wird nun auch noch gespeichert.

Möchte der Spielleiter aber alle Eingaben rückgängig machen, kann er nur auf [Cancel] klicken. Alle Änderungen werden ungeschehen gemacht. Im Hintergrund funktioniert das wie folgt: Bevor das Dialogfenster erstmals angezeigt wird, speichert der Rechner mit der Methode *saveStartUpValues()* alle aktuellen Extrapunkte der Teams in einem Vektor. Bricht nun der Spielleiter die Eingabe von Extrapunkten ab, werden wieder die alten Werte mit einer for-Schleife in die jeweiligen *TeamStatus* geladen.

5.2.8.2 Andere Antworten anzeigen / Grünes i

Hat ein Team eine Antwort zu einer Kreativaufgabe abgegeben, informiert der *CreativeTaskHandler* mit der Methode *sendMessageToAllClients(..)* die einzelnen PDAs. Diese setzen daraufhin im Hintergrund in der Klasse *CreativeTask* die Boolean-Variable *solutionsAvailable* auf TRUE. Öffnet nun ein Spieler, welcher die Aufgabe bisher noch nicht löste, diese Kreativaufgabe, wird der Button [Other Answers] rot eingefärbt. So wird der Benutzer aufmerksam gemacht, dass andere Teams diese Aufgaben schon gelöst haben und er sich von diesen Antworten inspirieren lassen kann.

Klickt nun der Spieler auf diesen Button, öffnet sich ein Fenster mit dem Text „Downloading...“. Im Hintergrund werden nun die Lösungen vom Server angefordert und heruntergeladen. Wurden alle Antworten vollständig transferiert, wird folgendes Fenster angezeigt:

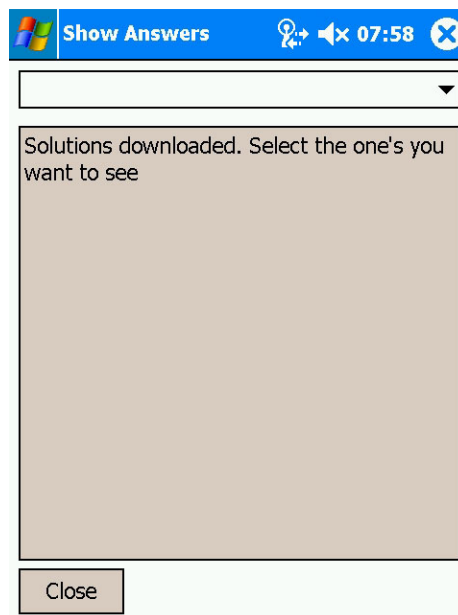


Abbildung 37: Andere Antworten anzeigen

Im Hintergrund wurde beim Aufrufen die Nachricht *GameEvents.NEW_CREATIVETASK_SOLUTION_REQUEST_EVENT* an den Server versandt. Der *CreativeTaskHandler* sendet nun mit der Methode *handleLoadAnswerRequest(..)* alle vorhandenen Lösungen an den Client. Müssen Annotationen übertragen werden, fordert der Client diese mit dem Event *GameEvents.NEW_CREATIVETASK_DATA_REQUEST_EVENT* beim Server an. Sobald alle Antworten und Annotationen übertragen wurden, ersetzt das System die Meldung im Textfenster. Der Spieler sieht nun, dass er die verschiedenen Antworten betrachten kann. Per Drop-Down-Menü wählt er das Team aus, dessen Lösung er sehen möchte. Schliesst er diese geöffnete Lösung, hat er die Möglichkeit andere Antworten auszuwählen und zu betrachten.

5.2.8.3 HTML-Export

Die bestehende Exportfunktion wurde erweitert. So können nun auch die Rangliste mit den Punktwerten und die Kreativaufgaben mit den Spielerlösungen exportiert werden. Um die Rangliste auszugeben wird beim *GameStatus* der Ranglisten-Vektor mit den Daten über die Methode *getRankingTable()* abgerufen. Dieser Vektor wird mit der Methode *writeRankingTable(..)* in HTML-Code eingebettet. Zuerst wird die Überschrift und anschliessend die einzelnen Einträge generiert. Am Ende sieht die Tabelle wie folgt aus:

Rankingtable		
Rank	Team	Points
1.	team6	25
2.	team1	23
3.	team2	19
4.	team4	18
5.	team5	14
6.	team3	11

Abbildung 38: Exportierte Rangliste

Auch die Lösungen der einzelnen Teams werden für eine Nachbesprechung exportiert. Wie der Abbildung 39 entnommen werden kann, wird zuerst die Aufgabenstellung und darauf folgend die Musterlösung angezeigt. Anschliessend werden die einzelnen Teamlösungen exportiert. Zuerst wird das Erstellerteam aufgeführt (bei einer Gruppenarbeit zusätzlich noch die anderen mitwirkenden Teams), dann die Bewertung und der Übersicht halber erst zum Schluss die Spielerantwort.

Distanz

Question	Misst die Distanz zwischen den Hörsälen 04-G-30 und 24-G-45 auf eine möglichst kreative Art und Weise. Wählt selber eine passende Antwortform und gibt die Distanz mit den Masseinheiten an. Lasst euch von den anderen Antworten inspirieren. PS: 20 Schritte oder 39 Schuhlängen sind un kreativ.
Solution	(keine Musterlösung)

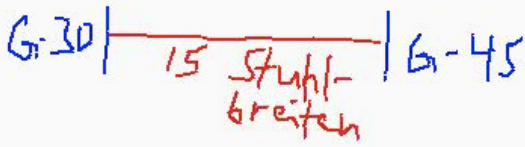
- team2	
Rank:	1.
Response:	

Abbildung 39: Exportierte Aufgabe mit Spielerlösung

6 Software-Tests

Während dem Implementieren der einzelnen Module wurde immer wieder deren Funktionstüchtigkeit getestet. Nun galt es diese so geprüften Komponenten systemweit zu prüfen. Das fertiggestellte System sollte daher einem gründlichen Test unterzogen werden. „Im Systemtest sollen möglichst viele derjenigen Fehler gefunden werden, die bei den [vorherigen] Tests übersehen wurden, weil sie nur beim Zusammenspiel zwischen den verschiedenen Komponenten auftreten.“ (Glinz WS 2005/2006, S. 118)

In dieser Diplomarbeit wurden drei Tests durchgeführt. Zuerst wurde mit einem Techniktest überprüft, ob die entwickelte Software im Gesamtverbund mit der an der Universität vorhandenen Infrastruktur und dem Positionierungssystem Ekahau funktionierte. Im zweiten Test testeten erstmals auch reale Benutzer diese neuen Module. Aufgrund deren Rückmeldung wurde die Software angepasst und noch benutzerfreundlicher gestaltet. Im letzten Test wurden diese Änderungen ein weiteres Mal überprüft.

6.1 Techniktest

Der Techniktest wurde von Christoph Göth und mir am 24. Oktober 2007 an der Binzmühlestrasse in Oerlikon (Institut für Informatik) durchgeführt. Wir testeten die Software mit zwei Geräten des Typs Qtek 9000 und zwei HP iPAQ 7400 - PDAs.

6.1.1 Aufgabe

Es galt die im Kapitel „Implementation“ aufgeführten Module auf allfällige Fehler zu überprüfen. Erstmals wurde das komplette System getestet. Das ganze Positionierungssystem wurde hinzu geschaltet. Die realen Positionen der einzelnen Teams (PDAs) konnten automatisch angezeigt werden und mussten nicht mehr vom Spielserver aus simuliert werden.

6.1.2 Durchführung

Da beim ersten Spieldurchgang noch die „SmokeMap“¹⁷ hinzu geschaltet war, wurden die Client-Rechner sehr schnell an ihre Kapazitätsgrenzen gebracht. Es entstanden unkontrollierte Fehler, welche nicht reproduzierbar waren. Das System wurde in einem zweiten Rundgang noch einmal mit angepassten Einstellungen neugestartet. Nun konnten alle neu programmierten Funktionen getestet werden. Da die Aufgabenstellungen diejenigen waren, welche auch beim Benutzertest 1 verwendet werden sollten, konnten wir diese nahezu 1:1 überprüfen.

6.1.3 Analyse, Erkenntnisse und durchgeführte Verbesserungen

Technisch gesehen gab es keine Probleme mit den neuen Modulen. Das Versenden sowie Herunterladen von Antworten anderer Spieler funktionierten tadellos. Auch bei den Kreativ- und Kettenaufgaben sind keine Fehler aufgetreten. Es wurden aber folgende Mängel erkannt:

- Wenn alle Aufgaben gelöst wurden, soll dem Spieler via Chatnachricht angezeigt werden, dass keine weiteren Aufgaben vorhanden sind. Dies ist nicht der Fall wenn Kettenaufgaben im Spiel verwendet werden. In der Clientsoftware wurde die Anzahl gelöster Aufgaben mit einem falschen Wert verglichen, darum entsprach die Anzahl absolvierter Aufgaben niemals der Anzahl vorhandener Aufgaben.

¹⁷ Nebelkarte: „Nebel“ deckt zu Beginn die komplette Spielkarte ab. Wenn der Spieler sich nun im Gebäude bewegt, wird jeweils das Kartenfeld um die aktuelle Position klar dargestellt. Verlässt der Spieler diese Position, bleibt die Karte aufgedeckt. So wird die sichtbare Karte immer grösser je länger das Spiel dauert und je mehr der Spieler sich bewegt. Diese Funktion wurde von Linard Moll mit der Bachelorarbeit „Awareness von Bewegungsmustern“ dem mExplorer hinzugefügt.

Dieser Fehler konnte behoben werden. Der Vergleich wurde angepasst und die Statusmeldung „Alle Tasks gelöst.“ wird nun auch bei der Verwendung von Kettenaufgaben angezeigt.

- Wenn auf dem Server eine Voice-Antwort abgespielt wird, öffnet sich automatisch der Windows Media Player. Soll nun eine weitere Voice-Lösung abgespielt werden, gibt es einen Datei-Zugriffskonflikt, weil der Server die Auslagerungsdatei überschreiben will, welche momentan vom Windows Media Player verwendet wird.

Dieser Fehler konnte insofern behoben werden, als dass dem Benutzer eine Fehlermeldung angezeigt wird, sollte dieser Konflikt vorliegen. Er wird dazu aufgefordert den Windows Media Player zu schliessen und dann die Voice-Antwort nochmals abzuspielen.

6.2 Benutzertest 1

Am Dienstagabend, 30. Oktober 2007, wurde der erste der beiden Benutzertests auf dem Campus Irchel durchgeführt. 8 Personen zwischen 23 und 31 Jahren nahmen daran teil. Zum Einsatz kamen wie im Technikertest zwei Geräte Qtek 9000 und zwei Geräte HP iPAQ 7400.

6.2.1 Aufgabe

Die Aufgaben dieses Tests waren vielfältig. Einerseits sollten die Module im echten Einsatz getestet und Fehler gefunden werden und andererseits sollte auch die Wirksamkeit der neuen Systemteile mit den bisherigen Aufgabenmodulen verglichen werden. Der Ablauf des Tests war wie folgt: Zuerst wurden die Spieler von mir instruiert. Ich erläuterte unter anderem die Bedeutung der einzelnen Symbole auf der Benutzeroberfläche und zeigte die wichtigsten Funktionen auf. Anschliessend sollten die Spieler zuerst zwei, drei Warm-Up-Aufgaben lösen um die „richtigen“ Aufgaben ohne Probleme zu absolvieren. Um die Erfahrungen der Spieler zu erfassen hatten die Benutzer ganz am Ende des Tests noch einen Fragebogen über das Erlebte zu beantworten. Die Aufgaben und Instruktionen sind im Anhang 8.2 aufgeführt. Der Fragebogen befindet sich im Anhang 8.3.

6.2.2 Durchführung

Bei dem Test hatten die Teams mit diversen Problemen zu kämpfen. Einerseits wurden die Positionen nicht immer zuverlässig angezeigt. So gab es teilweise Fälle, in denen der Positionspunkt immer zwischen zwei Orten hin und her sprang, welche gut 20 Meter auseinanderlagen und durch zwei Wände abgetrennt wurden. Andererseits froren vor allem die Qteks noch häufig ein. Die Spieler setzten dann jeweils die Geräte zurück und meldeten sich erneut beim Server an.

Zusätzlich gab es Fälle in denen die Übermittlung der Lösungen nicht auf Anhieb klappte. So erhielten die Geräte teilweise keine eindeutige ID vom Server, welche notwendig ist, damit Clients Annotationen übertragen können. Der Spieler musste also die Übertragung erneut starten. Das Herunterladen der Antworten der anderen Teams benötigt eine grosse Zeitspanne. Da die Meldung „Downloading...“ nicht verändert wurde, hatten viele Benutzer das Gefühl, der Rechner sei eingefroren, obwohl dies in Tat und Wahrheit nicht der Fall war. Alles in allem kann man sagen, dass die Abläufe der einzelnen Module funktionierten (die Teams konnten alle Aufgaben lösen) aber die Datentransfers optimiert werden sollten. So kann versucht werden die Wartezeiten zu verkürzen und dem Spieler wieder zu mehr Spass verhelfen.

6.2.3 Durchgeführte Verbesserungen

Da beim Qtek mit der Windows Mobile Version 5.0 der Tastatur-Button in der Mitte des unteren Bildschirmrandes angezeigt wird, verdeckt er die in der Mitte angeordneten Buttons des mExplorers. Es wurden nun die Koordinaten der Buttons so korrigiert, dass soweit möglich alle verdeckten Schaltflächen in

der linken, unteren Ecke des Displays angezeigt werden. Dies führt zu einer komfortableren Bedienbarkeit der Benutzeroberfläche.

Die vielen Datentransfers sind darauf zurück zu führen, dass mehr oder weniger unkontrolliert die Rangliste und die Anzahl Punkte des Teams versendet wurden (sprich zwei Nachrichten mit teilweise überschneidendem Inhalt). Bis anhin wurde immer wenn die Registerkarte „Tasks“ des Servers aktualisiert wurde auch die Rangliste an die Clients gesendet. Dies konnte eine sehr grosse Datenflut erzeugen. Dieses Problem wurde nun behoben. Es wird nur noch die Rangliste versendet. Daraus kann der Client seinen Punktestand und die aktuelle Position herauslesen und in der Spielerinformation anzeigen lassen. Mit dieser Massnahme konnte die Anzahl versendeter Nachrichten um bereits mehr als 50% reduziert werden. Zusätzlich wird die Rangliste nur noch beim Eintreffen der folgenden Ereignisse versandt:

- Extra-Points: der Spielleiter hat den Teams Extrapunkte verteilt.
- Nominalscale / Rating Score: Nach der Korrektur und Punktevergabe durch den Spielleiter wird die Rangliste neu berechnet und muss demzufolge wieder an die Spieler versandt werden.
- Normale Aufgabe: hat ein Team eine geschlossene Aufgabe gelöst, erhält es eine gewisse Anzahl Punkte. Aufgrund dieser neuen Punktzahl muss die Rangliste frisch berechnet werden. Anschliessend ist diese zu versenden.
- Fangen (Hunting-Catching): wurde ein Team gefangen, erhält der Fänger eine gewisse Anzahl Punkte. Die neue Rangliste muss dann an alle Teams verschickt werden.

6.3 Benutzertest 2

Am Donnerstag, 22. November 2007, wurde der zweite der beiden Benutzertests auf dem Campus Irchel durchgeführt. Diesmal nahmen 18 Personen zwischen 19 und 30 Jahren daran teil. Zum Einsatz kamen neun Geräte HP iPAQ 7400.

6.3.1 Aufgabe

Es wurden prinzipiell dieselben Aufgaben verwendet wie beim ersten Test. Dadurch soll es möglich sein die Wirksamkeit der getätigten Verbesserungen nach dem ersten Benutzertest zu überprüfen. Jedoch habe ich die vier Warm-Up-Aufgaben auf eine einzige reduziert. Ich konnte beim ersten Test beobachten, dass dies ausreicht um die Bedienung rudimentär zu üben. Zusätzlich habe ich die Telefonnummer-Aufgabe so angepasst, dass nur noch die letzte Ziffer gefragt wird. Beim ersten Test gab es insofern Probleme, dass einzelne Spieler die Landesvorwahl eingaben und andere nicht. Den Fragebogen habe ich prinzipiell so belassen. Nur eine Frage „Wie gut kannten Sie den Campus Irchel vor dem Spiel?“ wurde neu hinzugefügt.

6.3.2 Durchführung

Die Durchführung des Tests war schon gefühlsmässig mit weniger Problemen behaftet als der erste Test: Die Testpersonen kamen viel weniger zu mir um den PDA neu zu starten oder um mir vorhandene Fehler zu melden. Diese Reduktion der technischen Probleme kann weiter unten auch der Analyse der Fragebogen entnommen werden. Nur soviel vorne weg: Das Gefühl hat mich nicht getäuscht.

Das Problem des springenden Positionspunktes und dessen träge Verschiebung auf der Karte konnte nicht wirklich behoben werden. Als Workaround¹⁸ platzierte ich die Aufgaben dorthin, wo der Empfang gut war und der Punkt auch weniger hüpfte.

6.3.3 Durchgeführte Verbesserungen

Da das Lösen von Gruppenaufgaben nicht als sehr einfach empfunden wurde, habe ich die Software so angepasst, dass der Spieler einfacher die gewünschten Teams auswählen und kontaktieren kann (im Kapitel 5.2.3.1 beschrieben). Bisher musste ein Team zur Kontaktaufnahme immer den Chat verwenden. Chatnachrichten wurden aber eher selten gelesen, weil oft nicht bemerkt wurde, dass der PDA eine neue Chatmitteilung empfangen hat. Mit den Neuerungen wird den Spielern die Möglichkeit geboten auf eine einfache Art und Weise miteinander in Kontakt zu treten, damit die Teams auch sicher erkennen, dass ein anderes Team mit ihnen die Aufgabe lösen möchte. Das Popup¹⁹, welches neu eingeblendet wird, überdeckt alle anderen Anzeigen und die Spieler müssen es notgedrungen lesen oder zumindest wegklicken. Wie die aufgeforderten Teams sich dann verhalten, kann vom System schlecht gesteuert werden, da die Spieler nicht gezwungen werden können zum anderen Team zu gehen und mit diesem gemeinsam die Aufgabe zu lösen.

6.4 Analyse der Fragebogen

Zuerst werde ich die Fragebogen der einzelnen Tests auswerten. Anschliessend vergleiche ich die beiden Analysen miteinander, zeige Veränderungen auf und ziehe schlussendlich ein Fazit.

Um die Wirkung der Kreativaufgaben nachvollziehen und statistisch zeigen zu können, wurden im Test vier Kreativaufgaben und drei klassische Aufgaben eingesetzt. Bei der Kreativaufgabe „Gedicht“ sollten die Spieler ein Liebesgedicht an den Kopierer schreiben. Die Kettenaufgabe „Outdoor“ war zweiteilig: im ersten Teil galt es, die Brunnenskulptur vor der Mensa in kreativerweise zu erweitern und im zweiten Teil der Aufgabe mussten diese Erweiterungen noch kommentiert werden. Eine weitere Aufgabe bestand darin, die „Distanz“ zwischen den durch den Lichthof getrennten Hörsälen auf eine möglichst kreative Weise zu messen. Als letzte Kreativaufgabe mussten die Spieler in der Aufgabe „Porträt“ in einer Handwriting-Annotation das Porträt eines Teammitglieds aufzeichnen. Bei der ersten klassischen Aufgabe hatten die Spieler ein Buch in der Bibliothek zu recherchieren, während in der Toilettenaufgabe die Anzahl WC-Türen in einem Verbindungsgang gezählt werden mussten. Als letzte Aufgabe mussten die Spieler an der Infosäule noch die Telefonnummer von Dirk Froberg herausfinden und in den PDA eingeben.

Die Aufgaben werden durch die Spieler im Fragebogen beurteilt anhand der Faktoren Spass (wie viel Spass hatten die Spieler beim Lösen der Aufgabe), Schwierigkeitsgrad (wie anspruchsvoll war die Aufgabe), Inspiration (wie stark liessen sich die Spieler von den anderen Teams beeinflussen) sowie Nutzen (wie nützlich war die Aufgabe um den Campus kennen zu lernen).

Die detaillierte Übersicht über die ausgewerteten Fragebogen sind im Anhang 8.3 zu finden.

6.4.1 Analyse Fragebogen Benutzertest 1

Insgesamt hatten die Spieler Spass am Spiel (3.86, N = 8, Skala: 1-5) auch wenn es durch technische Probleme stark behindert wurde (5). Die Zeit ist mittelschnell verstrichen (3.29) und die Benutzer würden es begrüßen, wenn auch Museen und Zoos solche Lernspielumgebungen anbieten und verwenden würden

¹⁸ Workaround: „Unter einem Workaround versteht man allgemein eine Behelfslösung zur Umgehung eines nicht "sauber" lösaren Problems.“ (<http://www.lexitron.de/main.php?detail=true&eintrag=1165>)

¹⁹ Popup: „Popup-Windows sind Fenster [...] um dem Anwender wichtige Hinweise zu geben. Sie müssen erst weggeklickt werden, damit der Anwender weiterarbeiten kann.“ (<http://www.it-administrator.de/lexikon/popup.html>)

(3.86). Die Fragen empfanden sie als ausgewogen (3.14) und nicht zu schwierig aber auch nicht zu leicht. Dies wird weiter unten bei der Feinauswertung der beiden Aufgabentypen nochmals gezeigt. Ab und zu liessen sich die Spieler beim Antworten von den anderen Teams beeinflussen (2.71). Die Eingabe der normalen als auch der Gruppenantworten fanden die Spieler einfach (3.86 bzw. 3.43). Die Rangliste wurde mittelmässig verwendet (3.00), während sie aber nur einzelne Spieler zu besseren Leistungen anspornte (2.71). Nachdem eine Aufgabe gelöst wurde, interessierte es die Spieler nur selten wie die nachfolgenden Teams diese Aufgabe beantworteten (2.29).

Die nachfolgende Tabelle gibt eine Übersicht über die durchschnittlichen Bewertungen der Aufgaben.

Aufgabe	Gedicht	Outdoor	Kommentar	Distanz	Porträt	Bibliothek	Toiletten-türen	Info-säule
Spass	4.13	4.25	2.75	4.00	4.50	3.63	3.25	2.50
Schwierigkeitsgrad	3.50	2.63	3.63	3.50	2.75	3.38	1.75	3.13
Inspiration	3.63	3.75	3.13	3.88	4.00	3.43	2.75	2.88
Nutzen	2.13	3.38	2.88	2.88	3.13	3.50	3.25	3.38
Skala von 1 (sehr wenig) bis 5 (sehr viel), N = 8; Tiefster Wert Höchster Wert								

Tabelle 14: Ergebnisse zu den Aufgaben des Fragebogens vom ersten Benutzertest

Um die beiden Aufgabenarten optimal vergleichen zu können, wurde ein unabhängiger t-Test durchgeführt. Die folgenden Parameter wurden verwendet:

- 95%-Konfidenzintervall
- $\alpha = 0.05$ (Signifikanzniveau, maximal zulässige Irrtumswahrscheinlichkeit)²⁰
- Nullhypothese: „Kreativaufgaben und klassische Aufgaben haben denselben Spassfaktor, denselben Schwierigkeitsgrad, dieselbe Inspiration auf andere Spieler und denselben Nutzen.“

Die Kreativaufgaben erzeugten durchschnittlich einen höheren Spass beim Spieler (3.93, N = 40) als bei klassischen Aufgaben (3.13, N = 24). Dies wurde von den Spielern im Allgemeinen auch bestätigt, beantworteten sie doch die Frage „Kreativaufgaben haben mir viel mehr Spass bereitet als die anderen Aufgaben“ mit dem hohen Wert 3.86 (N = 8). Auch der durchgeführte unabhängige t-Test zeigt auf, dass der Spass beim Lösen von Kreativaufgaben signifikant höher ist als bei klassischen Aufgaben (Sig. (2-tailed) = 0.010 < α). Die Nullhypothese kann verworfen werden. Der Spassfaktor ist bei Kreativaufgaben höher.

Kreativaufgaben wurden von den Spielern durchschnittlich schwieriger (3.2, N = 40) empfunden als die klassischen Aufgaben (2.75, N = 40). Hier zeigt der durchgeführte unabhängige t-Test aber keinen signifikanten Unterschied zwischen den Aufgabentypen (Sig. (2-tailed): 0.127 > α). Daraus folgt, dass kein Aufgabentyp prinzipiell schwieriger ist als der andere.

Die Antworten der anderen Teams inspirierten die Spieler bei Kreativaufgaben durchschnittlich mit einem Wert von 3.68 (N = 40) während die Inspiration bei klassischen Aufgaben nur einen mässigen Wert von 3.00 erzielte (N = 23). Bei den klassischen Aufgaben erwartete ich einen tieferen Wert, haben doch die Spieler vom System aus keine Möglichkeit die Antworten der anderen zu betrachten. Es ist davon auszu-

²⁰ „Das Signifikanzniveau (engl.: significance) bezeichnet die Wahrscheinlichkeit, mit der im Rahmen eines „Hypothesentests“ die „Nullhypothese“ fälschlicherweise verworfen werden kann, obwohl sie eigentlich richtig ist (Fehler erster Art oder α -Fehler). Es wird daher auch als Irrtumswahrscheinlichkeit bezeichnet.“ <http://psydok.sulb.uni-saarland.de/volltexte/2004/268/html/node151.html>

gehen, dass die Teams sich während dem Spiel auf dem Gelände austauschten und daher dieser Wert zustande kam. Die Nullhypothese wird durch den unabhängigen t-Test verworfen (Sig. (2-tailed): $0.018 < \alpha$).

Der durchschnittliche, von den Spielern empfundene Nutzen von klassischen Aufgaben liegt zwar mit 3.38 ($N = 24$) wesentlich höher als bei den Kreativaufgaben (2.88, $N = 40$), aber dennoch kann kein signifikanter Unterschied zwischen den beiden Aufgabentypen festgestellt werden (Sig. (2-tailed): $0.151 > \alpha$). Der Nutzen von klassischen Aufgaben wird höher eingeschätzt, weil der Bezug zum Gelernten in direktem Zusammenhang mit der Aufgabe steht. Bei der Bibliotheksaufgabe wissen die Spieler nachher wie ein spezifisches Buch zu finden ist. Bei einer Kreativaufgabe ist der direkte Kontextbezug weniger sichtbar. Die Spieler erkennen nicht auf Anhieb, dass sie etwas beim Lösen der Aufgabe gelernt haben. Göth und Froberg gehen in ihren Arbeitsberichten davon aus, dass durch Kreativaufgaben eine emotionale Verknüpfung zum Beispiel zum Kopiererstandort entsteht und diese Erinnerung sich tiefgreifender bei den Spielern festsetzt, als dies bei klassischen Aufgaben möglich wäre. Der Lerneffekt von Kreativaufgaben verankert sich mehr im Unterbewusstsein des Spielers (Göth und Schwabe, *Designing Tasks for Engaging Mobile Learning* 2007).

6.4.2 Analyse Fragebogen Benutzertest 2

Im zweiten Benutzertest hat sich der Spass am Spiel leicht erhöht (von 3.86 auf 4.11, $N = 18$, Skala: 1-5). Dieser gute Wert wurde sicherlich auch davon beeinflusst, dass das Spiel viel weniger durch technische Probleme behindert wurde. In der Umfrage sank der Wert von hohen 5 Punkten um etwa die Hälfte auf 2.61. Die unter 6.2.3 getroffenen Massnahmen verfehlten also ihre Wirkung nicht. Dass die Zeit auch viel schneller verstrichen ist (+1.3) als beim ersten Test, hat vermutlich ebenso mit der Senkung der technischen Probleme zu tun. Die Fragen wurden wiederum als ziemlich ausgewogen empfunden (2.83). Die Spieler liessen sich praktisch gleich stark von den Antworten der anderen Teams beeinflussen wie beim ersten Test (2.61). Das Lösen von Aufgaben in Gruppen wurde von diesen Testern als komplizierter empfunden (-0.29) als von der ersten Testgruppe. Dies kann damit zu tun haben, dass die Vertrautheit im Umgang mit Computern tiefer war (4.22) als bei den ersten Probanden (4.71). Die Eingabe normaler Antworten wurde hingegen wesentlich einfacher empfunden als von der ersten Testgruppe (4.61, +0.75). Obwohl die Rangliste geringfügig weniger oft geöffnet wurde (2.67), spornte sie im gleichen Umfang an, wie dies bei der ersten Testgruppe festgestellt werden konnte. Ich denke, um ein weitaus höheres Anspornen zu erzielen, könnte der Spielleiter mit anderen Anreizen arbeiten. So könnte das beste Team als Siegerpreis beispielsweise einen CHF 50.- Gutschein erhalten. Die extrinsische Motivation könnte so stark gesteigert werden. Wiederum hatten die Spieler mehr Spass an den Kreativ- als an den anderen Aufgaben (3.39). Dies konnte auch bei der Analyse der einzelnen Aufgaben nachgewiesen werden.

Die nachfolgende Tabelle gibt eine Übersicht über die durchschnittlichen Bewertungen der Aufgaben.

Aufgabe	Gedicht	Outdoor	Kom- mentar	Distanz	Porträt	Bibliothek	Toiletten- türen	Info- säule
Spass	3.28	3.78	3.50	3.94	4.39	3.00	2.78	3.17
Schwierig- keitsgrad	3.28	2.72	3.11	3.17	2.89	3.78	2.11	3.33
Inspira- tion	2.56	2.88	2.78	2.67	2.33	2.17	2.00	2.67
Nutzen	2.33	2.50	2.61	2.83	1.94	3.67	3.11	3.33
Skala von 1 (sehr wenig) bis 5 (sehr viel), N = 18; Tiefster Wert Höchster Wert								

Tabelle 15: Ergebnisse zu den Aufgaben des Fragebogens vom zweiten Benutzertest

Wie schon beim ersten Benutzertest hatten die Spieler mehr Spass an Kreativaufgaben (3.78, N = 90) als an den klassischen Aufgaben (2.98, N = 54). Dies wurde auch von den Spielern im Fragebogen bestätigt (3.39, N = 18). Der unabhängige t-Test bestätigte das Ergebnis aus dem ersten Benutzertest: die beiden Aufgabentypen sind unterschiedlich. Die Kreativaufgaben bereiten den Spielern durchschnittlich einen höheren Spass (Sig. (2-tailed): 0.000 < a).

Die 18 Probanden dieses zweiten Benutzertests empfanden die Kreativaufgaben und klassischen Aufgaben gleichschwierig (3.03, 3.07). Auch der unabhängige t-Test zeigte keine Unterschiede zwischen den beiden Aufgabenarten an (Sig. (2-tailed): 0.825 > a).

In diesem Test betrug die Inspiration der anderen Teams auf die Spieler bei Kreativaufgaben durchschnittlich mit einem Wert von eher tiefen 2.64 (N = 89). Bei klassischen Aufgaben war die Inspiration erwartungsgemäss noch tiefer (2.28, N = 54). Die oben verworfene Nullhypothese wird durch den zweiten t-Test diesmal knapp nicht verworfen (Sig. (2-tailed): 0.087 > a).

Der von den Spielern empfundene Nutzen von klassischen Aufgaben liegt wesentlich höher (3.37, N = 54) als bei den Kreativaufgaben (2.44, N = 90). Der unabhängige t-Test ergab diesmal einen signifikanten Unterschied zwischen den beiden Aufgabentypen (Sig. (2-tailed): 0.000 < a). Die Erwartungen aufgrund der Erkenntnisse von den verschiedenen, von Christoph Göth durchgeführten Benutzertests wurden also vollumfänglich erfüllt.

6.5 Fazit

Den Spielern macht das spielende Lernen mit dem mExplorer Spass. Die neuartige, für die meisten unbekannte Technik ist interessant und will ausprobiert werden. Dieser Umstand erleichtert es für den Spielleiter die Spieler zu motivieren. Analysiert man die Auswirkungen der einzelnen Module, ist zu erkennen, dass die Lösungseingabe bei Kreativaufgaben als einfach empfunden wurde. Dies bestätigt das simpel gehaltene Eingabefenster (siehe Abbildung 18). Die Lösungseingabe bei Gruppenaufgaben wurde in beiden Tests schwieriger bewertet. Hier ist der Koordinationsaufwand zwischen den Teams vermutlich die grösste Schwierigkeit. Darum baute ich noch das Signalisierungsfenster ein (siehe Kapitel 5.2.3.1). Die Rangliste hatte in beiden Tests nur geringe Auswirkungen auf die Spieler. Es ist vorstellbar, dass die Spieler einfach den Spass am Lösen der Aufgaben hatten und sie weniger vom Spieltrieb „ich will der Beste sein“ geleitet wurden. Meine Meinung ist, dass dieser Wettbewerbsgedanke erst aufkommt, wenn die Bedienung des PDAs beherrscht wird, der Wow-Effekt über die Technik verschwunden ist und sich der Spielerfokus auf den Wettbewerb zwischen den einzelnen Teams verschoben hat. Ist dies der Fall, wird die Rangliste sicher öfters benutzt.

Die beiden Benutzertests zeigten auf, dass der Spassfaktor bei Kreativaufgaben klar höher ist als bei klassischen Aufgaben. Aber der entscheidende Faktor ist sicherlich der Spielleiter: Je innovativer er die Aufgaben stellt, desto höher ist der Spass, welcher die Spieler während des Spiels haben. Für die Komplexität und Schwierigkeit ist es egal, welcher Aufgabentyp eingesetzt wird. Kreativaufgaben können genauso schwierig sein wie klassische Aufgaben und umgekehrt. Der Faktor „Inspiration“ war beim ersten Test in Kreativaufgaben höher als in klassischen Aufgaben. Beim zweiten Test wurde die Nullhypothese knapp nicht verworfen. Hier könnte ein weiterer Test mit speziell auf die Inspiration zugeschnittenen Aufgaben Klarheit schaffen. Der Nutzen von klassischen Aufgaben wurde in beiden Benutzertests klar höher eingestuft als der Nutzen von Kreativaufgaben. Der zweite t-Test zeigte dann auch noch einen signifikanten Unterschied zwischen den Aufgabentypen auf. Es ist daher davon auszugehen, dass klassische Aufgaben einen offensichtlicheren, höheren Nutzen bringen als dies bei Kreativaufgaben der Fall ist. Es kann davon ausgegangen werden, dass der (versteckte) Erinnerungseffekt beim Ausfüllen des Fragebogens von den Spieler nicht beachtet wurde.

7 Schlusswort

Die erhaltene Aufgabenstellung konnte vollständig umgesetzt werden. Die didaktische Aufarbeitung des mExplorers konnte im Kapitel „Theoretische Einführung“ dargestellt werden. Im darauffolgenden Kapitel wurde zuerst die Anforderungsanalyse an die neuen Module beschrieben, welche dann anschliessend auch implementiert werden konnten. Die beiden Benutzertests zeigten klar die vorhandenen Fehler und technischen Schwierigkeiten auf. Während die vorhandenen Fehler beseitigt werden konnten, sind die technischen Probleme praktisch nicht zu beheben. Namentlich die Lokalisierung der Tags ist fehleranfällig und ungenau. Dies nimmt dem Spieler die Freude und die Motivation am Spiel. Wie in der didaktischen Aufarbeitung gezeigt, wird das Spiel dann beiseite gelegt.

Alles in allem darf ein positives Fazit gezogen werden, da die Ziele erreicht wurden und die neuen Funktionen in anderen Arbeiten und anderen Spielsituationen wiederverwendet werden können. Diese Module können auch noch weiter ausgebaut werden: So wäre es denkbar, dass sich alle Aufgaben in einer Kettenaufgabe befinden. Wird dann beispielsweise die erste Teilaufgabe gelöst, könnte nicht nur eine sondern mehrere nachfolgende Aufgaben auf dem Spielerbildschirm angezeigt werden. Dies würde es dem Spielleiter ermöglichen unterschiedliche Schwierigkeitsgrade ins Spiel zu bringen. Dem Spieler würde aber doch noch eine gewisse Freiheit bei der Auswahl der Lösungsreihenfolge der Aufgaben gelassen.

7.1 Persönliche Erfahrungen

Diese sechsmonatige Diplomarbeit war eine wertvolle Erfahrung für mich. Ich bekam die Aufgabenstellung und konnte Zeit und Abfolge der einzelnen Teilarbeiten selbst bestimmen. Obwohl schon viele Entwickler am mExplorer gearbeitet haben, konnte man sich doch an den Grundstrukturen der Architektur orientieren und wusste ungefähr, wo man ein Modul einsetzen musste und wo mit der Fehlersuche zu Beginnen ist. Da der mExplorer ein Prototyp ist, war mir von Beginn weg klar, dass die Einarbeitung in die Software Zeit beansprucht und eine Herausforderung ist. Bei Problemen jeglicher Art konnte ich mich immer an meinen Betreuer Christoph Göth wenden, welcher mir stets unkompliziert half, sofort Tipps gab und mögliche Lösungsansätze aufzeigte. Diese Arbeit hat mir sehr viel Spass und Freude bereitet.

8 Anhang

8.1 Aufgabenstellung

Spielregeln, Aufgabenarten und Bewertungsmethoden in mobilen Spielen

Aufgabenstellung der Diplomarbeit von Marcel Schönbächler

Ziel dieser Diplomarbeit ist es, die Spielgestaltung des mExplorers didaktisch aufzuarbeiten und Fragestellungen komplexeren Ausmasses zu ermöglichen. Bisher ist es nämlich nur möglich eingeschränkte Fragen wie Multiple-Choice-Fragen und ähnliche zu stellen, was die potentiellen Einsatzmöglichkeiten sehr begrenzt.

Der Ablauf der Diplomarbeit gestaltet sich wie folgt: nach einer ausgedehnten Literatureinarbeitung und Analyse von anderen bestehenden mobilen (Lern-)Spielen wird mit Christoph Göth und Dirk Froberg ein Experteninterview durchgeführt und so die Anforderungen für die praktische Umsetzung der Erkenntnisse erhoben.

In einem neuen Software-Modul soll der mExplorer um die Resultate der Literatureinarbeitung und Experteninterviews erweitert werden, so dass sich die Einsatzmöglichkeiten des mExplorers erhöhen. Jegliche Erweiterungen sind sauber in die bestehende Gesamtarchitektur zu integrieren.

Es stehen folgende Ideen zur Diskussion:

- Die momentan geschlossenen Aufgabenstellungen sollen erweitert werden, so dass kreative und offene Aufgaben gestellt werden können, welche während des Spiels direkt vom Spielleiter beurteilt werden können. Am Ende des Spiels sollen die Resultate mit einer Rangliste eingeblendet werden können.
- Es sollen „Kettenaufgaben“ im System eingesetzt werden können. Dies ist so eine Art moderne Schnitzeljagd, bei dem es darum geht, durch das sequentielle abarbeiten von Hinweisen zum gewünschten Ziel zu gelangen.
- Alle diese oben erwähnten Aufgabenstellungen sollen so implementiert werden, dass mehrere Teams gleichzeitig am Spiel teilnehmen können.
- Um alle diese Anforderungen implementieren zu können, muss der Editor (serverseitig) erweitert werden. Auch am Client sind die notwendigen Änderungen durchzuführen: so soll der Spieler jeweils den eigenen Punktestand angezeigt bekommen und zusätzlich noch die Punktestände der anderen Spieler sehen können. Am Ende des Spiels soll auch jeder Client die Rangliste mit den Resultaten dem Spieler präsentieren.

Um die vorgenommenen Änderungen am mExplorer zu testen, wird in einer ersten Phase ein Pretest mit rund 10 Testpersonen durchgeführt. Die beim Test gefundenen Mängel werden anschliessend gefiltert und verbessert.

Beim abschliessenden User-Test mit rund 20 Testpersonen werden die getätigten Änderungen nochmals überprüft. Durch Befragung, Beobachtung und einer Log-File-Auswertungen sollen in diesem Test die Nützlichkeit, Akzeptanz und die Usability der neuen Module getestet werden.

Die eigentliche Diplomarbeit umfasst neben einem Überblick über den mExplorer, ein ausführliches Literaturkapitel, in dem der theoretische Teil der Problemstellung abgehandelt wird. Den grössten Teil der Arbeit machen die technischen Dokumentationen der neuen Module und die Ergebnisse der Benutzertests aus.

8.2 Benutzertests - Instruktionen und Aufgaben

Die Benutzertests sind in zwei Phasen aufgeteilt: In der Warm-Up-Phase geht es darum, dass die Spieler die Software kennenlernen. Diese Aufgaben sollen als Ketten-Aufgabe durchgeführt werden, so wird sichergestellt, dass jeder Spieler die Grundlagen der Bedienung gelernt hat. In der zweiten Phase werden dann die eigentlichen Spielaufgaben gestellt.

8.2.1 Instruktionen

Damit der Test in geordneten Bahnen ablaufen kann, sind folgende Anweisungen zu beachten:

- Wenn eine Aufgabe mit dem Partnerteam zu lösen ist, dann gilt folgendes:
 - Partnerteams sind wie folgt einander zugeteilt:
 - team1 - team2
 - team3 - team4
 - team5 - team6
 - ...
 - Nur ein Team sendet die Antwort. Das nicht-sendende Team wird vom Sender über den Button [Groupwork] angewählt, damit die Lösung für beide Teams gezählt wird.
 - Ist das Partnerteam nicht gerade in der Nähe, schliesst die Aufgabe [Cancel] und nimmt mit dem anderen Team via Chat Kontakt auf. Wenn beide Teams vor Ort sind, öffnet wieder die Aufgabe und löst sie.
- Wenn eine Aufgabe geöffnet wird und der [Other Answers]-Button rot eingefärbt ist, klickt ihn an und lasst euch von den Antworten der anderen Teams inspirieren.
- Um Komplikationen zu vermeiden, schickt die Antwort nur ab, wenn der PDA auch mit dem WLAN verbunden ist (Sendemasten-Symbol links von der Uhr). Andernfalls kann es sein, dass eine Antwort ein zweites Mal eingegeben werden muss.
- Eine permanente Aufgabe ist, möglichst viele andere Teams zu fangen. Zu fangende Teams sind mit einem Schaf gekennzeichnet. Euer Jäger wird durch ein Wolf-Symbol dargestellt.

8.2.2 Warm-Up-Phase

In einer Kettenaufgabe „Warm-Up“ sollen die folgenden Aufgaben gelöst werden. „Einstellungen“ geben die zu verwendenden Einstellungen bei der Erfassung der Kreativaufgabe an.

1. Text-Aufgabe
Schreibt in einer [Text]nachricht eure beiden Namen auf und schickt sie ab.
Einstellungen: Rating Score
2. Hand-Aufgabe
Löst die folgende Aufgabe mit eurem Partnerteam.
Aufgabenstellung:
Schreibt eine [Hand]writing-Lösung: schreibt euren Teamnamen und den des Partnerteams auf.
Einstellungen: Nominalscale, GroupWork
3. Photo-Aufgabe
Macht ein Eigenporträt von euch beiden mit der [Photo]kamera. Hat euer PDA keine Fotokamera integriert, gibt eine beliebige Antwort ein.
Einstellungen: Nominalscale
4. Voice-Aufgabe
Liest zuerst diese Instruktionen durch:
 1. Schliesst nach dem Lesen diese Aufgabe.

2. Wählt das „grüne i“ an wo die Photo-Aufgabe war. Betrachtet dort die Fotos der anderen Teams.
3. Öffnet diese Aufgabe wieder und erstellt eine [Voice]-Antwort. Sagt, wieviele Fotos ihr betrachtet haben könntet.
4. Klickt nach dem Senden auf die Rangliste. Jetzt sollte der Spielleiter schon die ersten Punkte verteilt haben.

Einstellungen: Rating Score

Anpassungen für den zweiten Benutzertest:

Es wird nur noch eine Warm-Up-Aufgabe verwendet. Aufgrund der gesammelten Erfahrung im ersten Benutzertest reicht dies aus.

8.2.3 Spielphase

Während der Spielphase werden die folgenden Kreativaufgaben gestellt. Die Aufgaben 1 und 4 wurden von Christoph Göth's Feldversuchen übernommen (Göth und Schwabe, Designing Tasks for Engaging Mobile Learning 2007), während die Aufgaben 5 und 6 der Diplomarbeit von Christoph Wirz entnommen wurden. Die Abbildungen von Seite 86 und 87 zeigen die Positionen der Aufgaben im Gelände an.

Einzelaufgabe

1. Gedicht

Erfindet mit eurem Partner-Team ein zweizeiliges Liebesgedicht an den Kopierer und sendet es als Text-Antwort ab. Wichtige Bedingung: es muss sich reimen.

Einstellungen: Nominalscale, GroupWork; *Map:* Hoersaele

Kettenaufgabe „Outdoor“

2. Brunnen

„Erweitert“ diese Brunnenskulptur: stellt euch passend zu dieser Skulptur hin und lasst euch von einem Passanten fotografieren. Inspiriert euch zuvor an den anderen Antworten. Habt ihr keine Fotokamera, skizziert wie ihr die Skulptur erweitert.

Einstellungen: Nominalscale; *Map:* Aussenanlage

3. Kommentar

Kommentiert eure Erweiterung: erzählt, was diese Skulptur und was ihr darstellt.

Einstellungen: Rating Score; *Map:* Aussenanlage

Einzelaufgabe

4. Distanz

Misst die Distanz zwischen den Hörsälen 04-G-30 und 24-G-45 auf eine möglichst kreative Art und Weise. Wählt selber eine passende Antwortform und gibt die Distanz mit den Masseinheiten an. Lasst euch von den anderen Antworten inspirieren. PS: 20 Schritte oder 39 Schuhmäßen sind un kreativ.

Einstellungen: Rating Score; *Map:* Hoersaele

Einzelaufgabe

5. Porträt

Erstellt eine [Hand]writing-Lösung: Zeichnet das Porträt eines Teammitglieds.

Einstellungen: Rating Score; *Map:* Hoersaele

Kettenaufgabe „Rundgang“

6. Bibliothek

Recherchiert das Buch „Agency and the Semantic Web“ vom Autor „Christopher Walton“ an einem Recherchecomputer. Gebt das erste Wort oben links von Seite 75 in Kleinbuchstaben ein.

Einstellungen: Open Question Task; *Map:* Strickhof; *Lösung:* participant

Anpassung für den zweiten Benutzertest:

- *Neue Frage:* Recherchiert das Buch „Programmieren mit Java“ vom Autor „Reinhard Schiedermeier“ an einem Recherchecomputer. Gebt den Titel des Themas von Seite 153 in Kleinbuchstaben und ohne Leerschläge ein.
- *Lösung:* konstanten

7. Toiletten

Wie viele Toilettentüren sind in diesem Durchgang zu finden?

Einstellungen: Slider Task: 0-10; *Map:* Hoersaele; *Lösung:* 4

8. Infosäule

Wie lautet die Telefonnummer von Dirk Froberg? (Gebt sie ohne Leerzeichen ein.)

Einstellungen: Open Question Task; *Map:* Hoersaele; *Lösung:* 0446356712

Anpassung für den zweiten Benutzertest:

- *Neue Frage:* Wie lautet die letzte Ziffer der Telefonnummer von Dirk Froberg?
- *Lösung:* 2

8.2.3.1 Übersicht über die Aufgaben

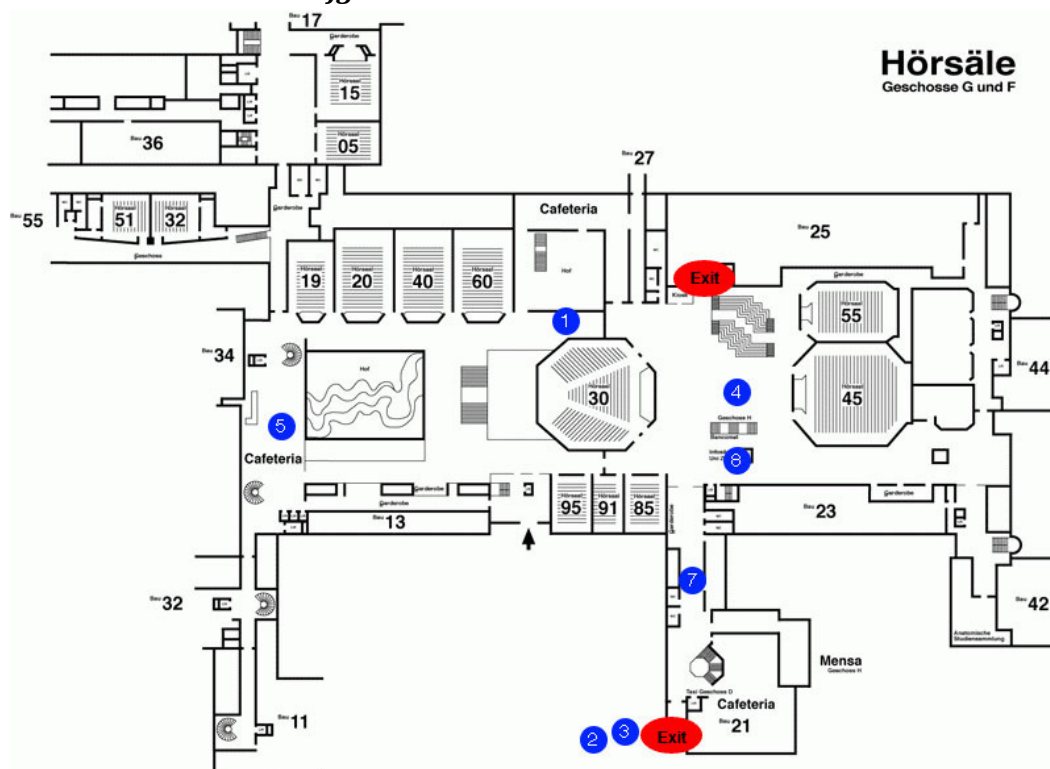


Abbildung 40: Übersicht über die verschiedenen Aufgaben innerhalb des Campus Irchel (Karte: Hoersaele.gif)



Abbildung 41: Kettenaufgabe "Outdoor" (Karte: Aussenanlage.gif)



Abbildung 42: Interview-Aufgabe im Strickhof (Karte: Strickhof.gif)

8.3 Auswertungen der Fragebogen

8.3.1 Ausgewerteter Fragebogen Benutzertest 1

Persönliche Daten

1. Teamnummer	
2. Geburtsjahr (JJJJ)	1981
3. Geschlecht	Männlich Weiblich

Fragen zum persönlichen Profil

1. Wie vertraut sind Sie im Umgang mit Computergeräten?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> (4.71) X <input type="checkbox"/>	Sehr
2. Wie vertraut waren Sie vor dem Spiel im Umgang mit Handhelds/ PDA?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> X (2.29) <input type="checkbox"/>	Sehr

Fragen zum allgemeinen Eindruck

1. Wie viel Spass hat Ihnen die Teilnahme am Spiel insgesamt gemacht?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.86) X <input type="checkbox"/>	Sehr viel
2. Wie viel haben Sie allgemein bei dem Spiel für sich profitiert?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.86) <input type="checkbox"/>	Sehr viel
3. Wie hilfreich war das Spiel für Sie um den Campus kennen zu lernen?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> X (2.43) <input type="checkbox"/>	Sehr gut
4. Ist die Zeit beim Spielen langsam oder schnell verstrichen?	Sehr langsam	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (3.29) <input type="checkbox"/>	Sehr schnell
5. Wie gerne würden Sie mit Hilfe eines solchen Spiels weitere Umgebungen kennen lernen, z. B. ein Museum, einen Zoo oder eine Stadt.	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.86) X <input type="checkbox"/>	Sehr gerne
6. Wie stark hätte eine 10 minütige Unterbrechung (z.B. ein Anruf oder Ausfall des Netzes) im Spiel gestört?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (4.29) X <input type="checkbox"/>	Sehr stark
7. Wie sehr wurde das Spiel durch technische Probleme behindert?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> (5.00) X <input type="checkbox"/>	Sehr stark
8. Wie motivierend und spassfördernd war für Sie allgemein die eingesetzte Technik?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (4.29) X <input type="checkbox"/>	Sehr stark

Fragen zum Lösen der Aufgaben

1. Die Aufgaben waren allgemein...	zu leicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (3.14) <input type="checkbox"/>	zu schwierig
2. Ich habe oft vor dem Lösen die Antworten der anderen Teams betrachtet. (Bei Aufgaben wo es möglich war.)	nie	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.71) <input type="checkbox"/>	immer
3. Das Lösen der Aufgaben in Gruppen war...	zu umständlich	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (3.29) <input type="checkbox"/>	einfach
4. Die Erfassung einer Gruppenantwort auf dem PDA war...	zu kompliziert	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (3.43) <input type="checkbox"/>	einfach
5. Die Erfassung einer normalen Antwort auf dem PDA war grundsätzlich...	zu kompliziert	-- - 0 + ++ <input type="checkbox"/> (3.86) X <input type="checkbox"/>	einfach

6. Ich habe die Rangliste häufig geöffnet.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (3.00) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
7. Die Rangliste spornte mich an.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.71) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
8. Es interessierte mich, nachdem ich die Aufgabe gelöst habe zu sehen, wie die Aufgabe von den anderen Teams gelöst wurden.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> X (2.29) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
9. Kreativaufgaben (offene Aufgaben) haben mir viel mehr Spass bereitet als die anderen Aufgaben.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> (3.86) X <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich

Fragen zu den Aufgaben im Spiel

Bitte geben Sie für jede der Aufgaben in der untenstehenden Tabelle an, a) wie viel Spass sie Ihnen gemacht hat, b) wie anspruchsvoll die Aufgabe für Sie war, c) wie stark die Lösungen der anderen Teams Sie beeinflussten, inspirierten sowie d) welchen Nutzen die Aufgabe hat um den Campus Irchel kennenzulernen:

1. Gedicht Erfindet mit eurem Partner-Team ein zweizeiliges Liebesgedicht an den Kopierer.				
Spass	Sehr wenig	-- - 0 + ++ <input type="checkbox"/> (4.13) X <input type="checkbox"/>	Sehr viel	Kommentare: - Ortung funktionierte nicht, Partner-Team hat Aufgabe gelöst.
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++ <input type="checkbox"/> (3.50) X <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++ <input type="checkbox"/> (3.63) X <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++ <input type="checkbox"/> X (2.13) <input type="checkbox"/>	Starker Nutzen	

Kettenaufgabe „Outdoor“

2.1 Brunnen „Erweitert“ diese Brunnenskulptur: stellt euch passend zu dieser Skulptur hin und lasst euch von einem Passanten fotografieren. Lasst euch von den anderen Antworten inspirieren. Habt ihr keine Fotokamera, skizziert, wie ihr diesen Brunnen erweitert.				
Spass	Sehr wenig	-- - 0 + ++ <input type="checkbox"/> (4.25) X <input type="checkbox"/>	Sehr viel	Kommentare: - Handheld ohne Kamera -> gezeichnet
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++ <input type="checkbox"/> X (2.63) <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++ <input type="checkbox"/> (3.75) X <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++ <input type="checkbox"/> X (2.75) <input type="checkbox"/>	Starker Nutzen	

2.2 Kommentar					
Kommentiert eure Erweiterung: erzählt, was diese Brunnenskulptur und was ihr darstellt.					
Spass	Sehr wenig	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (2.75)	Sehr viel	Kommentare:
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	<input type="checkbox"/> (3.63) X <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (3.13)	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (2.88)	Starker Nutzen	

3. Distanz					
Misst die Distanz zwischen den Hörsälen 04-G-30 und 24-G-45 auf eine möglichst kreative Art und Weise.					
Spass	Sehr wenig	-- - 0 + ++	<input type="checkbox"/> (4.00) X <input type="checkbox"/>	Sehr viel	Kommentare: - Lustig!
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	<input type="checkbox"/> (3.50) X <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++	<input type="checkbox"/> (3.88) X <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (2.88)	Starker Nutzen	

4. Porträt					
Erstellt eine [Hand]writing-Lösung: Zeichnet das Porträt eines Teammitglieds.					
Spass	Sehr wenig	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> (4.50) X	Sehr viel	Kommentare: - Senden hat nicht funktioniert („No ID received“)
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (2.75)	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++	<input type="checkbox"/> (3.75) X <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (3.13)	Starker Nutzen	

Kettenaufgabe „Rundgang“

5. Bibliothek					
Recherchiert das Buch. Gebt das erste Wort ein. (Richtige Lösung: participant)					
Spass	Sehr wenig	-- - 0 + ++	<input type="checkbox"/> (3.63) X <input type="checkbox"/>	Sehr viel	Kommentare: - Orientierung in Bibliothek schwer -> Übergang von Outdoor-Karte angeben
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (3.38)	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++	<input type="checkbox"/> <input type="checkbox"/> X (3.43)	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++	<input type="checkbox"/> (3.50) X <input type="checkbox"/>	Starker Nutzen	

5.2 Toilettentüren					
Wie viele Toilettentüren sind in diesem Gang zu finden?					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare:	
		<input type="checkbox"/> <input type="checkbox"/> X (3.25)			
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
		<input type="checkbox"/> X (1.75) <input type="checkbox"/>			
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
		<input type="checkbox"/> <input type="checkbox"/> X (2.75)			
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		
		<input type="checkbox"/> <input type="checkbox"/> X (3.38)			

5.3 Infosäule					
Wie lautet die Telefonnummer von Dirk Frohberg?					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Nummer je nach Maske mit oder ohne +41	
		<input type="checkbox"/> <input type="checkbox"/> X (2.50)			
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
		<input type="checkbox"/> <input type="checkbox"/> X (3.13)			
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
		<input type="checkbox"/> <input type="checkbox"/> X (2.63)			
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		
		<input type="checkbox"/> <input type="checkbox"/> X (3.13)			

Weitere Kommentare

Hier hat es Platz für Bemerkungen:

(Was hat (nicht) gut funktioniert? Was könnte besser gestaltet werden? Ist etwas zu unübersichtlich?)

- An und für sich eine coole Sache, es hapert einfach an der „Hardware“, das nimmt einem die Freude am Spiel. Wir wurden auf unserer Karte selten dort gefunden wo wir standen, deshalb hat es sich ein wenig in die Länge gezogen.
- Buttons sind oft hinter Tastatur-Aufklappbutton
- Ortung im Aussenbereich auf dem Weg zur Bibliothek funktioniert nicht
- Hilfreich wäre, auf der Karte scrollen zu können
- Software mehrmals abgestürzt (vor allem beim Downloaden)
- Positionsbestimmung nicht optimal
- Technik des Basissystems muss dringend optimiert werden

Vielen Dank für deine Teilnahme an diesem Test!

Es wäre toll dich auch beim Schlusstest 22. November 2007 1630 Campus Irchel begrüßen zu dürfen.

8.3.2 Ausgewerteter Fragebogen Benutzertest 2

Persönliche Daten

1. Teamnummer	
2. Geburtsjahr (JJJJ)	1982
3. Geschlecht	Männlich Weiblich

Fragen zum persönlichen Profil

1. Wie vertraut sind Sie im Umgang mit Computergeräten?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (4.22) X <input type="checkbox"/>	Sehr
2. Wie vertraut waren Sie vor dem Spiel im Umgang mit Handhelds/ PDA?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> X (2.39) <input type="checkbox"/>	Sehr

Fragen zum allgemeinen Eindruck

1. Wie viel Spass hat Ihnen die Teilnahme am Spiel insgesamt gemacht?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (4.11) X <input type="checkbox"/>	Sehr viel
2. Wie viel haben Sie allgemein bei dem Spiel für sich profitiert?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.00) X <input type="checkbox"/>	Sehr viel
3.1 Wie gut kannten Sie den Campus Irchel vor dem Spiel?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> X (2.78) <input type="checkbox"/>	Sehr gut
3.2 Wie hilfreich war das Spiel für Sie um den Campus kennen zu lernen?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.44) X <input type="checkbox"/>	Sehr gut
4. Ist die Zeit beim Spielen langsam oder schnell verstrichen?	Sehr langsam	-- - 0 + ++ <input type="checkbox"/> (4.56) X <input type="checkbox"/>	Sehr schnell
5. Wie gerne würden Sie mit Hilfe eines solchen Spiels weitere Umgebungen kennen lernen, z. B. ein Museum, einen Zoo oder eine Stadt.	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.94) X <input type="checkbox"/>	Sehr gerne
6. Wie stark hätte eine 10 minütige Unterbrechung (z.B. ein Anruf oder Ausfall des Netzes) im Spiel gestört?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (3.17) X <input type="checkbox"/>	Sehr stark
7. Wie sehr wurde das Spiel durch technische Probleme behindert?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> X (2.61) <input type="checkbox"/>	Sehr stark
8. Wie motivierend und spassfördernd war für Sie allgemein die eingesetzte Technik?	Gar nicht	-- - 0 + ++ <input type="checkbox"/> (4.06) X <input type="checkbox"/>	Sehr stark

Fragen zum Lösen der Aufgaben

1. Die Aufgaben waren allgemein...	zu leicht	-- - 0 + ++ <input type="checkbox"/> X (2.83) <input type="checkbox"/>	zu schwierig
2. Ich habe oft vor dem Lösen die Antworten der anderen Teams betrachtet. (Bei Aufgaben wo es möglich war.)	nie	-- - 0 + ++ <input type="checkbox"/> X (2.61) <input type="checkbox"/>	immer
3. Das Lösen der Aufgaben in Gruppen war...	zu umständlich	-- - 0 + ++ <input type="checkbox"/> (3.00) X <input type="checkbox"/>	einfach
4. Die Erfassung einer Gruppenantwort auf dem PDA war...	zu kompliziert	-- - 0 + ++ <input type="checkbox"/> (3.28) X <input type="checkbox"/>	einfach
5. Die Erfassung einer normalen Antwort auf dem PDA war grundsätzlich...	zu kompliziert	-- - 0 + ++ <input type="checkbox"/> (4.61) X <input type="checkbox"/>	einfach

6. Ich habe die Rangliste häufig geöffnet.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.67) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
7. Die Rangliste spornte mich an.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.72) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
8. Es interessierte mich, nachdem ich die Aufgabe gelöst habe zu sehen, wie die Aufgabe von den anderen Teams gelöst wurden.	überhaupt nicht	-- - 0 + ++ <input type="checkbox"/> X (2.28) <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich
9. Kreativaufgaben (offene Aufgaben) haben mir viel mehr Spass bereitet als die anderen Aufgaben.	überhaupt nicht	-- - 0 + ++ (3.39) X <input type="checkbox"/> <input type="checkbox"/>	stimmt vollumfänglich

Fragen zu den Aufgaben im Spiel

Bitte geben Sie für jede der Aufgaben in der untenstehenden Tabelle an, a) wie viel Spass sie Ihnen gemacht hat, b) wie anspruchsvoll die Aufgabe für Sie war, c) wie stark die Lösungen der anderen Teams Sie beeinflussten, inspirierten sowie d) welchen Nutzen die Aufgabe hat um den Campus Irchel kennenzulernen:

1. Gedicht Erfindet mit eurem Partner-Team ein zweizeiliges Liebesgedicht an den Kopierer.				
Spass	Sehr wenig	-- - 0 + ++ (3.28) X <input type="checkbox"/> <input type="checkbox"/>	Sehr viel	Kommentare: - Lustige Idee - Ein Beispiel wäre bei einer solchen Frage gut - Eine eher doofe Aufgabe - Gute Idee um Standort Kopierer kennenzulernen
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++ (3.28) X <input type="checkbox"/> <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.56) <input type="checkbox"/> <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++ <input type="checkbox"/> X (2.33) <input type="checkbox"/> <input type="checkbox"/>	Starker Nutzen	

Kettenaufgabe „Outdoor“

2.1 Brunnen „Erweitert“ diese Brunnenskulptur: stellt euch passend zu dieser Skulptur hin und lasst euch von einem Passanten fotografieren. Lasst euch von den anderen Antworten inspirieren. Habt ihr keine Fotokamera, skizziert, wie ihr diesen Brunnen erweitert.				
Spass	Sehr wenig	-- - 0 + ++ <input type="checkbox"/> (3.78) X <input type="checkbox"/>	Sehr viel	Kommentare: - Mit Fotokamera hätte es sicher mehr Spass gemacht. Handy und dann Bluetooth? - kreativ
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.72) <input type="checkbox"/>	Sehr schwer	
Inspiration	Kein Einfluss	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.88) <input type="checkbox"/>	Starker Einfluss	
Nutzen	Kein Nutzen	-- - 0 + ++ <input type="checkbox"/> <input type="checkbox"/> X (2.50) <input type="checkbox"/>	Starker Nutzen	

2.2 Kommentar Kommentiert eure Erweiterung: erzählt, was diese Brunnenskulptur und was ihr darstellt.					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Gut, dass man auch handschriftlich antworten kann. - Beschreibung einfach	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

3. Distanz Misst die Distanz zwischen den Hörsälen 04-G-30 und 24-G-45 auf eine möglichst kreative Art und Weise.					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Gute Kreativeaufgabe - gut	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

4. Porträt Erstellt eine [Hand]writing-Lösung: Zeichnet das Porträt eines Teammitglieds.					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Schwierig mit Handwriting zu zeichnen. - Wir sind nicht so die PDA-Zeichner - gut	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

5. Bibliothek Recherchiert das Buch. Gebt das erste Wort ein. (Richtige Lösung: konstanten)					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - mühsam: Buch war nicht da	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

5.2 Toilettentüren					
Wie viele Toilettentüren sind in diesem Gang zu finden?					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Gute Aufgabe für Neulinge - nützlich	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

5.3 Infosäule					
Wie lautet die letzte Ziffer der Telefonnummer von Dirk Froberg?					
Spass	Sehr wenig	-- - 0 + ++	Sehr viel	Kommentare: - Haben zuerst an Plakaten geschaut, bis wir den Infopunkt gesehen haben. - Gute Aufgabe	
Schwierigkeitsgrad	Sehr leicht	-- - 0 + ++	Sehr schwer		
Inspiration	Kein Einfluss	-- - 0 + ++	Starker Einfluss		
Nutzen	Kein Nutzen	-- - 0 + ++	Starker Nutzen		

Weitere Kommentare

Hier hat es Platz für Bemerkungen:

(Was hat (nicht) gut funktioniert? Was könnte besser gestaltet werden? Ist etwas zu unübersichtlich?)

- Haben mehr auf die PDA-Karte geschaut. Leider zeigte es oft nicht an wo wir oder die anderen waren
- Positionsangabe war etwas träge
- Punkt zu träge
- "Automatischer Kartenwechsel hat nicht funktioniert.
- Wolf/Schaf hat toll geklappt.
- Tipp: Weg zum Strickhof auf der Karte einzeichnen."
- Dauert relativ lange bis Verschiebung angezeigt wird. Während Aufgabe lösen, sehe ich nicht wenn Wolf kommt.
- Problem war, dass die Karte / Zielpunkt nur 2D dargestellt ist. Bsp. Bibliothek: Wir waren in einem falschen Stockwerk, konnten aber trotzdem eine Antwort auf die Aufgabe geben
- Die Zeichnungsfunktion finde ich toll; Ja / Nein-Fragen sollen auch auf diese Weise gelöst werden
- z. T. Sprünge beim Standort
- war ziemlich lässig, aber wir haben nie jemanden erfolgreich gejagt
- Automatischer Kartenwechsel hat nicht funktioniert. Für nicht ortskundige schwierig den Strickhof zu finden, da nicht auf Aussenanlage markiert.

Vielen Dank für deine Teilnahme an diesem Test!

8.3.3 Mit SPSS ausgewerteter Benutzertest 1

Mittelwerte

AufgabenNr		Spass	Schwierigkeit	Inspiration	Nutzen
1	Mean	4.1250	3.5000	3.6250	2.1250
(Gedicht)	N	8	8	8	8
2	Mean	4.2500	2.6250	3.7500	3.3750
(Outdoor)	N	8	8	8	8
3	Mean	2.7500	3.6250	3.1250	2.8750
(Komm.)	N	8	8	8	8
4	Mean	4.0000	3.5000	3.8750	2.8750
(Distanz)	N	8	8	8	8
5	Mean	4.5000	2.7500	4.0000	3.1250
(Porträt)	N	8	8	8	8
6	Mean	3.6250	3.3750	3.4286	3.5000
(Biblot.)	N	8	8	7	8
7	Mean	3.2500	1.7500	2.7500	3.2500
(Toilet.)	N	8	8	8	8
8	Mean	2.5000	3.1250	2.8750	3.3750
(Infos.)	N	8	8	8	8
Total	Mean	3.6250	3.0313	3.4286	3.0625
	N	64	64	63	64

Tabelle 16: Mittelwerte der verschiedenen Aufgaben

Übersicht

	Aufgabentyp	N	Mean	Std. Deviation	Std. Error Mean
Spass	1	40	3.9250	1.20655	.19077
	2	24	3.1250	1.11560	.22772
Schwierigkeit	1	40	3.2000	1.01779	.16093
	2	24	2.7500	1.29380	.26410
Inspiration	1	40	3.6750	.91672	.14495
	2	23	3.0000	1.27920	.26673
Nutzen	1	40	2.8750	1.32409	.20936
	2	24	3.3750	1.34528	.27460

Tabelle 17: Mittelwerte der Faktoren der beiden Aufgabentypen

Aufgabentyp: Kreativaufgaben (1); Klassische Aufgaben (2);

Unabhängiger t-Test

	Levene's Test for Equality of Variances		t-test for Equality of Means						
	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Spass	.136	.713	2.640	62	.010	.80000	.30303	.19425	1.40575
Schwierigkeit	2.320	.133	1.545	62	.127	.45000	.29127	-.13224	1.03224
Inspiration	1.617	.208	2.429	61	.018	.67500	.27786	.11938	1.23062
Nutzen	.016	.899	-1.454	62	.151	-.50000	.34392	-1.18748	.18748

Tabelle 18: Unabhängiger t-Test Benutzertest 1

8.3.4 Mit SPSS ausgewerteter Benutzertest 2

Mittelwerte

AufgabenNr		Spass	Schwierigkeit	Inspiration	Nutzen
1 (Gedicht)	Mean	3.2778	3.2778	2.5556	2.3333
	N	18	18	18	18
2 (Outdoor)	Mean	3.7778	2.7222	2.8824	2.5000
	N	18	18	17	18
3 (Kom.)	Mean	3.5000	3.1111	2.7778	2.6111
	N	18	18	18	18
4 (Distanz)	Mean	3.9444	3.1667	2.6667	2.8333
	N	18	18	18	18
5 (Porträt)	Mean	4.3889	2.8889	2.3333	1.9444
	N	18	18	18	18
p6 (Biblot.)	Mean	3.0000	3.7778	2.1667	3.6667
	N	18	18	18	18
7 (Toilet.)	Mean	2.7778	2.1111	2.0000	3.1111
	N	18	18	18	18
8 (Infos.)	Mean	3.1667	3.3333	2.6667	3.3333
	N	18	18	18	18
Total	Mean	3.4792	3.0486	2.5035	2.7917
	N	144	144	143	144

Tabelle 19: Mittelwerte der verschiedenen Aufgaben

Übersicht

	Aufgabentyp	N	Mean	Std. Deviation	Std. Error Mean
Spass	1	90	3.7778	.96893	.10213
	2	54	2.9815	1.12419	.15298
Schwierigkeit	1	90	3.0333	1.01062	.10653
	2	54	3.0741	1.16314	.15828
Inspiration	1	89	2.6404	1.23626	.13104
	2	54	2.2778	1.18825	.16170
Nutzen	1	90	2.4444	1.02892	.10846
	2	54	3.3704	1.35000	.18371

Tabelle 20: Mittelwerte der Faktoren der beiden Aufgabentypen

Aufgabentyp: Kreativaufgaben (1); Klassische Aufgaben (2)

Unabhängiger t-Test

	Levene's Test for Equality of Variances		t-test for Equality of Means						
	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Spass	.572	.451	4.493	142	.000	.79630	.17723	.44594	1.14665
Schwierigkeit	2.107	.149	-.221	142	.825	-.04074	.18420	-.40486	.32338
Inspiration	.000	.985	1.726	141	.087	.36267	.21017	-.05283	.77817
Nutzen	5.376	.022	-4.640	142	.000	-.92593	.19954	-1.32037	-.53148

Tabelle 21: Unabhängiger t-Test Benutzertest 2

8.4 Bibliographie

8.4.1 Literaturverzeichnis

- Ammann, Roland. *swissgeocache.ch - Die Cachetypen*. 2006. http://www.swissgeocache.ch/page_1_4.html (Zugriff am 08. August 2007).
- Anderson, R.C., Kulhavy, R. W. & André, T. «Feedback procedures in programmed instruction.» *Journal of Educational Psychology*, 62, 1971: 148-156.
- Brewster, Jean, Gail Ellis, und Denis Girard. *The Primary English Teacher's Guide*, 5. Auflage. Edinburgh, Harlow, Essex: Pearson Education Limited, 2004.
- Brunner, Dennis. *Architekturentwicklung für mobile Spiele*. Zürich: Institut für Informatik, Universität Zürich, 2005.
- Bütler, Monika. «Lernen ohne Üben?» *Das Beste (Reader's Digest)*, Dezember 2007: 13-16.
- Cheung, Priscilla. *Charles River City: An Educational Augmented Reality Simulation Pocket PC Game*. Massachusetts Institute of Technologie - Department of Electrical Engineering and Computer Science, 2003, 95-98.
- Deutsch, B., Deutsch C., Grondel S., Schaffrath G. *Roledges HomePage - Was ist ein Rollenspiel?* 22. Februar 1997. <http://www.geocities.com/TimesSquare/Alley/3003/d.definition.html> (Zugriff am 09. August 2007).
- Dougiamas, Martin. *Moodle - A Free, Open Source Course Management System for Online Learning*. <http://www.moodle.org> (Zugriff am 08. August 2007).
- Glinz, Prof. Dr. Martin. *Software Engineering - Eine Einführung*. Zürich: Universität Zürich, WS 2005/2006.
- Göth, Christoph. *Prototypische Implementierung einer mobilen Spielumgebung für den PDA*. Universität Koblenz-Landau, 2003.
- Göth, Christoph, Dirk Frohberg, und Gerhard Schwabe. «Von passivem zu aktivem mobilen Lernen.» *Zeitschrift für E-Learning*, 4 | 2007.
- Göth, Christoph, Gerhard Schwabe, und Dirk Frohberg. «The Focus Problem in Mobile Learning.» *IEEE 4th International Workshop on Wireless, Mobile and Ubiquitous Technologies in Education, Athens, Greece*, 2006.
- Göth, Christoph, und Dirk Frohberg. *"mExplorer"*. Arbeitsbericht der Arbeitsgruppe Informationsmanagement, Institut für Informatik, Universität Zürich, 2008.
- . *mExplorer - Arbeitsberichte*. Arbeitsgruppe Informationsmanagement, Institut für Informatik, Universität Zürich, 2007.
- Göth, Christoph, und Gerhard Schwabe. *Designing Tasks for Engaging Mobile Learning*. Universität Zürich: (in Vorbereitung), 2007.
- Gronlund, Norman E. *Assessment of Students Achievement*. Bd. 6th. Boston: Allyn & Bacon, 1998.
- Host-Party.com, LLC - Detektiv- und Krimi-Spiele. *Krimispiele Games and Parties*. 2006. <http://de.host-party.com/howtohost.asp> (Zugriff am 09. August 2007).
- IfI, Institut für Informatik. «Prüfungs- und Promotionsordnung für das Diplomstudium und das Doktorat in Informatik an der Wirtschaftswissenschaftlichen Fakultät der Universität Zürich.» 26. Februar 2001. http://www.ifi.uzh.ch/fileadmin/site/teaching/Reglemente/PPO01/ppo_informatik_2001.pdf (Zugriff am 08. August 2007).

Jacobs, Bernhard. *Guidelines zur Erstellen von Short-Answer-Aufgaben*. 04. April 2003. <http://www.phil.uni-sb.de/FR/Medienzentrum/verweise/psych/aufgaben/shortguide.htm> (Zugriff am 07. August 2007).

—. *Guidelines zur Konstruktion von Essay-Test-Aufgaben*. 29. März 2000. <http://www.phil.uni-sb.de/FR/Medienzentrum/verweise/psych/aufgaben/essayguide.htm> (Zugriff am 07. August 2007).

—. *Guidelines zur Konstruktion von Multiple-Aufgaben-Aufgaben*. 21. Oktober 2005. <http://www.phil.uni-sb.de/FR/Medienzentrum/verweise/psych/aufgaben/mcguideline.html> (Zugriff am 07. August 2007).

—. *Übungsaufgaben stellen mit JavaScript*. 06. Juli 2005. <http://www.phil.uni-sb.de/FR/Medienzentrum/verweise/psych/aufgaben/aufgaben.html#aufgabentypen> (Zugriff am 07. August 2007).

Klopfer, Eric, Jude Perry, Kurt Squire, Ming-Fong Jan, und Constance Steinkuehler. «Mystery at the Museum - Collaborative Game for Museum Education.» In *Computer Supported Collaborative Learning 2005: Proceedings of the 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years!*, 316 - 320. Taipei, Taiwan: Lawrence Erlbaum Associates Inc,US, 2005.

Krampe, G. «Effekte von Lehrerkommentaren zu Noten bei Schülern.» In *Fördernde Leistungsbeurteilung*, von R. Olechowski / E. Persy (Hrsg), 197 - 227. Wien / München, 1987.

Lefrancois, Guy R. *Psychologie des Lernens*. Berlin, Heidelberg: Springer, 1994, S. 3.

Lehrstuhl für Wirtschaftsinformatik, TU München. *CIO Business Game*. <http://www.cio-simulation.eu/web/guest/home> (Zugriff am 09. August 2007).

Lütgert, Prof. Dr. Will. *Leistung und Didaktik - Noten- und Berichtszeugnisse*. 18. Juni 2004. http://www.didaktik.uni-jena.de/did_05/noten.htm (Zugriff am 08. August 2007).

Luyten, Kris et al. «Designing for interaction: socially-aware museum handheld guides.» http://research.edm.uhasselt.be/~kris/research/publications/nodem2006/ARCHIE_-_Designing_for_interaction_-_final_version.pdf (Zugriff am 27. Juli 2007).

Menzel, Holger. *Feedbackinsatz in computerunterstützten Lernumgebungen mittels Macromedia Flash*. 30. August 2005. http://wwwstud.rz.uni-leipzig.de/~wir02doz/selfmadestuff/rest_study_ha01.htm (Zugriff am 08. August 2007).

Mietzel, Gerd. *Wege in die Entwicklungspsychologie, 4. Auflage*. Weinheim: Verlagsgruppe Beltz, Psychologie Verlags Union, 2002.

Moll, Linard. *Awareness von Bewegungsmustern*. Zürich: Institut für Wirtschaftsinformatik der Universität Zürich, 23.09.2007.

Moskal, Barbara M. *Scoring Rubrics? what, when and how?* 29. März 2000. <http://pareonline.net/getvn.asp?v=7&n=3> (Zugriff am 08. August 2007).

Prensky, Marc. *Digital Game-Based Learning*. New York: McGraw-Hill Education, 2001.

Sacher, Werner. *Prüfen – beurteilen – benoten, Grundlagen, Hilfen und Denkanstöße für alle Schularten*. Bad Heilbrunn: Klinkhardt, 1996.

Schmitt, Günter. *Skript Lernen und Verhaltensänderung*. Essen: Universität Essen, 1999, S.1.

Schulmeister, Rolf. *Lernplattformen für das virtuelle Lernen. Evaluation und Didaktik*. München, Wien: Oldenbourg, 2003.

Schwabe, G., Göth, C., Froberg, D. «Does Team Size Matter in Mobile Learning?» *International Conference on Mobile Business 2005*, 2005.

Schweizer Armee, Chef der Armee. *Ausbildungsmethodik, Reglement 51.018d*. Schweizer Armee, 2005.

Van Loon, et al. *Archives & Museum Informatics: Museums and the Web 2007: Papers: Van Loon, et al., Supporting Social Interaction: A Collaborative Trading Game On PDA*. 28. März 2007. <http://www.archimuse.com/mw2007/papers/vanLoon/vanLoon.html> (Zugriff am 09. August 2007).

Wikipedia. *Rollenspiel (Spiel)*. 27. Juli 2004. http://de.wikipedia.org/wiki/Rollenspiel_%28Spiel%29 (Zugriff am 09. August 2007).

Wirz, Christoph. *Annotationen im "Mobile Learning Game"*. Zürich, 2007.

Yatani, Koji, et al. «Musex: A System for Supporting Children's Collaborative Learning in a Museum with PDAs.» *Systems and Computers in Japan*, Vol. 35, No. 14., 2004.

ZFH, Zürcher Fachhochschule. «Rahmenstudienordnung für Nachdiplomstudien ZFH.» 22. November 2005. http://www.ifu.zhwin.ch/wb/pdf/zhcm_rahmenstudienordnung.pdf (Zugriff am 08. August 2007).

Zimbardo, Philip G. *Psychologie*. Berlin, Heidelberg: Springer, 1992, S. 227.

8.4.2 Abbildungsverzeichnis

Abbildung 1: Ausschnitt Geo-Caching Karte "Nordost-Schweiz"	15
Abbildung 2: Der Lernprozess (eigene Darstellung)	25
Abbildung 3: Übersicht „Schriftliche Fragestellungen“ (eigene Darstellung)	26
Abbildung 4: Moodle-Logo	35
Abbildung 5: Der Lernprozess (eigene Darstellung)	40
Abbildung 6: Module Phase 1 (eigene Darstellung)	40
Abbildung 7: Module Phase 2 (eigene Darstellung)	42
Abbildung 8: Module Phase 3 - 1 (eigene Darstellung)	43
Abbildung 9: Module Phase 3 - 2 (eigene Darstellung)	44
Abbildung 10: Module Phase 4 (eigene Darstellung)	45
Abbildung 11: UML-Klassendiagramm: <i>ChainTask.java</i>	51
Abbildung 12: Erfassung einer Kettenaufgabe.....	51
Abbildung 13: Anzeige von Kettenaufgaben im Spieleditor.....	52
Abbildung 14: Anzeige eines Tasks einer Kettenaufgabe auf dem Client	53
Abbildung 15: Sequenzdiagramm: Serverseitige Behandlung von Kettenaufgaben	54
Abbildung 16: Erfassung einer Kreativaufgabe im Spieleditor	55
Abbildung 17: UML-Klassendiagramm: <i>CreativeTask.java</i>	56
Abbildung 18: Kreativaufgabe auf dem PDA angezeigt	56
Abbildung 19: UML-Klassendiagramm: <i>CreativeTaskResponse.java</i>	58
Abbildung 20: Sequenzdiagramm: Übertragen einer Spielerantwort (Übersicht).....	59
Abbildung 21: Sequenzdiagramm: Übertragung einer Spielerantwort (Client)	61
Abbildung 22: UML-Klassendiagramm <i>CreativeTaskHandler.java</i>	62
Abbildung 23: Erhalt einer CreativeTaskResponse: Verarbeitung im CreativeTaskHandler	62
Abbildung 24: Auswahl der beteiligten Teams	64
Abbildung 25: Groupwork-Request: Anfrage eines anderen Teams zum gemeinsamen Lösen einer Gruppenaufgabe	64
Abbildung 26: Übersicht über das Versenden von Groupwork-Requests.....	65
Abbildung 27: Sequenzdiagramm: Versenden von Groupwork-Request (Sicht Requestor).....	65
Abbildung 28: Sequenzdiagramm: Weiterleiten von Groupwork-Requests (Sicht Server)	66
Abbildung 29: Sequenzdiagramm: Empfangen einer Groupwork-Request (Sicht Recipient)	66
Abbildung 30: Erweiterung des CreativeTaskHandlers um die Gruppenarbeit.....	67

Abbildung 31: Übersicht über die empfangenen Antworten.....	69
Abbildung 32: Rating Score-Dialogfenster.....	69
Abbildung 33: Nominalskala-Dialogfenster	70
Abbildung 34: Übersicht über die verschiedenen Teams	71
Abbildung 35: Anzeige der Rangliste auf dem Client	72
Abbildung 36: Extrapunkte - Dialogfenster	72
Abbildung 37: Andere Antworten anzeigen.....	73
Abbildung 38: Exportierte Rangliste	74
Abbildung 39: Exportierte Aufgabe mit Spielerlösung.....	74
Abbildung 40: Übersicht über die verschiedenen Aufgaben innerhalb des Campus Irchel (Karte: Hoersaele.gif).....	86
Abbildung 41: Kettenaufgabe "Outdoor" (Karte: Aussenanlage.gif)	87
Abbildung 42: Interview-Aufgabe im Strickhof (Karte: Strickhof.gif).....	87

8.4.3 Tabellenverzeichnis

Tabelle 1: Lerninhalt, Lernaktivität und geeignete Spielarten.....	14
Tabelle 2: Beispielfrage für eine Multiple-Choice-Aufgabe.....	27
Tabelle 3: Beispielfrage für eine Short-Answer-Aufgabe	28
Tabelle 4: Beispielfrage für eine Essay-Test-Aufgabe	29
Tabelle 5: Notenübersicht aus (IfI 2001) und (ZFH 2005).....	31
Tabelle 6: Beispiel Scoring Rubric. In Anlehnung an http://upload.wikimedia.org/wikipedia/en/5/59/Graded_rubric.jpg (zuletzt geöffnet am 08. August 2007).....	32
Tabelle 7: Übersicht Stärken / Schwächen der Module der Phase 1.....	47
Tabelle 8: Übersicht Stärken / Schwächen der Module der Phase 2.....	47
Tabelle 9: Übersicht Stärken / Schwächen der Module der Phase 3.....	48
Tabelle 10: Übersicht Stärken / Schwächen der Module der Phase 4.....	48
Tabelle 11: Erläuterung der Felder bei der Erfassung von Kreativaufgaben	55
Tabelle 12: Erläuterung der Buttons bei einer Kreativaufgabe.....	57
Tabelle 13: Ereigniskennungen für die Übertragung einer Spielerlösung.....	59
Tabelle 14: Ergebnisse zu den Aufgaben des Fragebogens vom ersten Benutzertest	79
Tabelle 15: Ergebnisse zu den Aufgaben des Fragebogens vom zweiten Benutzertest.....	81
Tabelle 16: Mittelwerte der verschiedenen Aufgaben	96
Tabelle 17: Mittelwerte der Faktoren der beiden Aufgabentypen	96
Tabelle 18: Unabhängiger t-Test Benutzertest 1	96
Tabelle 19: Mittelwerte der verschiedenen Aufgaben	97
Tabelle 20: Mittelwerte der Faktoren der beiden Aufgabentypen	97
Tabelle 21: Unabhängiger t-Test Benutzertest 2	97