

**Hochschule für Technik und Wirtschaft Dresden (FH)
Fachbereich Informatik/Mathematik**

Diplomarbeit
im Studiengang Medieninformatik

**Virtuelle Welten auf dem iPhone – Konzeption und
Implementierung einer Reise durch virtuelle Welten
als interaktives Video**

eingereicht von: Ellen Krüger (21043)

eingereicht am: 17.11.2009

Betreuer: Prof. Dr. Teresa Merino

Inhaltsverzeichnis

Motivation und Ziel der Arbeit	5
1. Grundlegende Begriffe	7
1.1 Lernen	7
1.2 Wissen	8
1.3 Lernparadigmen	10
1.3.1. Der Behaviorismus	10
1.3.2. Der Kognitivismus	11
1.3.3. Der Konstruktivismus	13
1.4 Der Begriff virtuell im Kontext Bildung	14
1.5 Zusammenfassung	15
2. Klassifizierung virtueller Welten	17
2.1 Geteilter Platz / Raum	18
2.1.1. Veranschaulichungsprinzipien nach Schwan & Bruder	18
2.1.2. Sharable Content – Inhalte gemeinsam nutzen	22
2.2 grafische Nutzeroberfläche	25
2.2.1. 2D und 2,5D	25
2.2.2. 3D	27
2.2.3. Die Navigation in 2D- und 3D-Welten	28
2.3 Direktheit / Unmittelbarkeit	28
2.3.1. Definition Avatar	28
2.3.2. Aufgaben eines Avatar	29
2.3.3. Eingesetzte Technologien	34
2.3.4. Vor- und Nachteile des Einsatzes virtueller Charaktere im Lernprozess	40
2.4 Interaktivität	42
2.4.1. Stufe I: Objekte betrachten und rezipieren	43
2.4.2. Stufe II: Multiple Darstellungen betrachten und rezipieren	43
2.4.3. Stufe III: Die Repräsentationsform variieren	44

2.4.4.	Stufe IV: Inhalte der Komponenten beeinflussen	45
2.4.5.	Stufe V: Objekt bzw. Inhalt der Repräsentation konstruieren und Prozesse generieren	45
2.4.6.	Konstruktive und manipulierende Handlungen mit situationsabhängigen Rückmeldungen	46
2.5	Persistenz	46
2.6	Anwendung der Charakteristika zur Typisierung von Lernprogrammen ..	47
2.6.1.	Hypermedia-Systeme	48
2.6.2.	(intelligente) Tutoringsysteme und Übungsprogramme	49
2.6.3.	Simulationen	51
2.6.4.	Lernkonferenzsysteme	53
2.6.5.	Edutainment-Programme	55
2.6.6.	Soziale Welten	58
2.7	Zusammenfassung – Mehrwert virtueller Lernwelten	58
3.	Lernen via iPhone	61
3.1	Mobile Learning	61
3.2	Das iPhone	63
3.3	Die iPhone SDK	67
3.3.1.	Installation der iPhone SDK	68
3.3.2.	Xcode	69
3.3.3.	Cocoa Touch	71
4.	Umsetzung eines Media-Players für das iPhone	73
4.1	Vorüberlegungen	73
4.1.1.	Videos und Animationen	74
4.1.2.	Verwaltung der Inhalte	74
4.1.3.	Aufbau der XML-Datei	75
4.2	Das Model-View-Controller Design	78
4.2.1.	Model des Players	79
4.2.2.	View des Players	80
4.2.3.	Controller des Players	82

5. Zusammenfassung	83
5.1 Fazit der Diplomarbeit	83
5.2 Erweiterung des Players	84
Abkürzungen	86
Glossar	87
Abbildungs- und Tabellenverzeichnis	90
Literaturverzeichnis	91
Anlage I: Sammlung virtueller Welten	97
Selbstständigkeitserklärung	112

Motivation und Ziel der Arbeit

In der heutigen Zeit wird es immer schwerer, insbesondere für große und internationale Unternehmen, aus der Vielzahl der Möglichkeiten geeignete Methoden zur firmeninternen und auch -externen Kommunikation zu finden. Ein Weg sind virtuelle Welten. Die Vorteile liegen klar auf der Hand – zeitliche und räumliche Unabhängigkeit und trotzdem ein gemeinsamer Treffpunkt. Spannt man den Bogen weiter, erkennt man schnell die Möglichkeit solche Welten mit Lerninhalten zu füllen und für Aus- und Weiterbildungen, sowie Produktschulungen und ähnliches zu nutzen. Doch was macht eine Lernumgebung zur (virtuellen) Welt? Warum eine virtuelle Welt anstelle eines herkömmlichen Lern-Management-Systems? Was ist der Mehrwert und rechtfertigt dieser den Mehraufwand an Arbeit und vor allem Kosten für die Produktion aufwendiger Anwendungen?

Virtuelle Lernwelten sollen eine Atmosphäre schaffen, die alle Sinne anregt und das Lernen zu einem Erlebnis werden lässt. Inhalte werden zueinander in Beziehung gesetzt, indem abgeschlossene Themen in (virtuellen) Räumen gebündelt sind. Der Nutzer betritt eine Welt, kann navigieren und auf dem Weg bereits Informationen sammeln und anfassen.

Werfen wir einen Blick in die Zukunft. Nicht das Lernen an sich verändert sich, sondern vielmehr die genutzten Medien. Wurde früher mit Buch, Zettel und Stift gelernt, ist der Computer als Lernmedium aus der heutigen Zeit nicht mehr weg zu denken. Was ist der nächste Schritt? Können Lerninhalte in Zukunft auch auf mobilen Endgeräten zugänglich gemacht werden?

Ziel der Arbeit ist es, einen Überblick über virtuelle Welten zum Lernen zu bekommen. Dazu werden im ersten Kapitel vorab grundlegende Begriffe zum Thema Lernen und Wissen erläutert. Nachdem damit die Basis gebildet wurde, geht es im zweiten Teil vor allem um die Charakteristika virtueller Lernwelten. Es werden fünf Merkmale untersucht und im Anschluss anhand derer eine mögliche Gruppierung virtueller Welten vorgenommen.

Das dritte und vierte Kapitel beschäftigen sich dann mit einer Weiterführung des eLearning, dem mobile Learning. Kapitel 3 dient zur Vorstellung der Thematik und eines mobilen Endgerätes als Lernmedium, dem iPhone. Weiterhin wird die Entwicklungsumgebung vorgestellt, mit deren Hilfe Applikationen für das iPhone geschrieben werden können. Das bildet die Grundlage für Kapitel 4, der Umsetzung eines Media-Player zur Wiedergabe von Lerninhalten für das iPhone.

Im fünften und letzten Kapitel werden die Ergebnisse der Arbeit zusammengefasst. Außerdem wird ein Ausblick gegeben, um welche Funktionen der Player erweitert werden könnte um noch besser im Bereich mobile Learning einsetzbar zu sein.

1. Grundlegende Begriffe

Lernen oder das Erlangen von Wissen ist das vorrangige Ziel virtueller Lernwelten. Deshalb sollen diese Begriffe und darauf aufbauend Lernprinzipien und -paradigmen im ersten Teil der vorliegenden Arbeit näher betrachtet werden, um mehr darüber zu erfahren wie man lernt und welche Wirkung bestimmte Lernmethoden erzielen. Außerdem befasst es sich näher mit dem Begriff virtuell im Kontext virtuelles Lernen.

1.1 Lernen

Für den Begriff des Lernens findet man zahlreiche Definitionen in Lexika und anderen wissenschaftlichen Arbeiten. So beschreibt LexiROM (1999) das Lernen als *„das An-eignen von Wissen und Kenntnissen bzw. das Einprägen in das Gedächtnis. Das Lernen beinhaltet auch den Vorgang, im Laufe der Zeit durch Erfahrungen, Einsichten o.Ä. zu Einstellungen und Verhaltensweisen zu gelangen, die von Bewusstsein und Bewusstheit bestimmt sind.“*

Eine eher biologische Sichtweise wählen Zimbardo & Gerrig (1999), die sich auf die Reiz-Reaktions-Beziehung berufen, welche das Zusammenspiel zwischen Körper und Umwelt über die Sinnesorgane beschreibt. Weiterhin spricht Zimbardo (1992) von Verhaltensänderungen, die durch den Lernprozess herbeigeführt werden und bereits erlangte Erkenntnisse und Erfahrungen als Basis nutzen.

Auch Meyers Lexikon (1997) nutzt Reiz und Reaktion um den Lernprozess zu beschreiben. Danach ist das Resultat von Lernen *„[...] entstandene Verhaltensänderungen und -möglichkeiten, die Organismen befähigen, aufgrund früherer und weiterer Erfahrungen situationsangemessen zu reagieren.“* Außerdem wird die Wissensaneignung ein aktiver und überwiegend einsichtiger Prozess genannt.

An dieser Stelle soll noch eine letzte Definition genannt werden, da sie einen guten Zusammenhang zwischen aktiven und eher passiven Lernens darstellt.

"In unserer modernen Gesellschaft erfolgen [...] Lernprozesse vor allem in zwei Formen:

- *durch tätige Anteilnahme an dem Leben sozialer Gemeinschaften (Familie, Freundeskreis, Gleichaltrigengruppe),*
- *durch gezielte Eingriffe („Interventionen“) von erwachsenen Berufspädagogen (Kindergärtner, Lehrer, Sozialpädagogen, Freizeitpädagogen)“*

(Giesecke ,2001).

All die gefundenen Begriffserklärungen stimmen in einem wichtigen Faktor überein, der Erfahrung. Erfahrung sammelt der Mensch unbewußt, vorallem im Alltag. Sie ist sowohl Resultat, als auch die Basis für ein bestimmtes Verhalten. Das heißt, dass sich Einstellung und Verhalten durch und mit der Erfahrung ändern. Man lernt auf Situationen zu reagieren und sie zu bewältigen. Hinzu kommt das bewußte Lernen, das vorallem in Schule, Studium und Aus- und Weiterbildungen erzielt wird. Alle Formen des Lernens führen zur Aneignung neuen Wissens.

1.2 Wissen

Hier werden vorallem 3 Arten unterschieden, deklaratives, prozedurales und implizites Wissen.

Deklaratives - , auch statisches - oder Faktenwissen genannt, wird durch Sprache oder Bild repräsentiert. Auch wenn es statisch ist, ist es nicht zeitlich invariant. Das bedeutet, dass Wissen, das als richtig angenommen wird, jedoch falsch ist, durch plausible Erklärung widerlegt werden kann. Während dessen wird die neue, richtige Erkenntnis abgespeichert. Es ersetzt jedoch nicht das falsche Wissen. Dieses ist noch Wochen später als Erinnerung an einen Irrtum im Gedächtnis. Typische Beispiele für Faktenwissen bietet die Mathematik, wie zum Beispiel das $1 \text{ mal } 1$. Man weiß, drei mal drei ergibt neun, ohne eine Erklärung dafür zu brauchen.

Die zweite Art beschreibt das Wissen darüber, wie etwas abläuft oder funktioniert.

Hierbei handelt es sich um Wissen über bestimmte Prozeduren oder Verarbeitungsprozesse, die ein gewünschtes Resultat erzielen. Das Wissen ist dynamisch, das heißt, dass es durch neue Verarbeitungstechniken und -erweiterungen ständig angepasst wird. Prozedurales Wissen ist zielgerichtet, ein Gesamtziel lässt sich meist in Teilziele zerlegen. Die Wahl und Beschreibung der zur Umsetzung dieser Teilziele erforderlichen Werkzeuge oder Operationen entnehmen wir unserem prozeduralen Wissen. Basis dafür ist häufig Faktenwissen.

Auch hier kann ein Beispiel aus der Mathematik angewandt werden, die schriftliche Multiplikation. Ziel ist es komplexe Zahlen zu multiplizieren. Durch die Zerlegung in einzelne kleine Rechnungen gelangt man zum Gesamtergebnis. Grundlage für die schriftliche Multiplikation ist die Beherrschung des 1 mal 1.

Die komplexeste Form des Wissens ist das implizite Wissen. Zu dieser Gruppe zählen die motorischen Fähigkeiten, sowie das Erkennen und Verstehen. Beispielsweise weiß man, dass wenn man auf einen Teller mit Kartoffeln zeigt und in Verbindung damit die Bezeichnung laut ausspricht, nicht der Geruch oder die Farbe gemeint ist, sondern die Kartoffeln an sich. In den meisten Fällen ist diese Wissensform nicht verbalisierbar, weil es vor allem um den Verstehensprozess und die damit verbundene Bewußtseins-erweiterung geht. Grundlage hier ist deklaratives und prozedurales Wissen.

(vgl. Baumgartner, Payr 1994; Haag 1995)

1.3 Lernparadigmen

Im Wesentlichen werden 3 Lerntheorien oder -muster unterschieden, Behaviorismus, Kognitivismus und Konstruktivismus. Einen Überblick über die Lerntheorien und ihre Merkmale soll die nachstehende **Tabelle 1.1** bieten, bevor sie im Folgenden näher erläutert werden.

Tabelle 1.1: Überblick über die Lernparadigmen (Baumgartner & Payr 1994)

Kategorie	Behaviorismus	Kognitivismus	Konstruktivismus
Hirn ist ein	Passiver Behälter	Informationsverarbeitendes „Gerät“	Informationell geschlossenes System
Wissen wird	Abgelagert	Verarbeitet	Konstruiert
Wissen ist	Eine korrekte Input-Outputrelation	Ein adäquater interner Verarbeitungsprozeß	Mit einer Situation operieren zu können
Lernziele	Richtige Antworten	Richtige Methoden zur Antwortfindung	Komplexe Situationen bewältigen
Paradigma	Stimulus-Response	Problemlösung	Konstruktion
Strategie	Lehren	Beobachten und helfen	Kooperieren
Lehrer ist	Autorität	Tutor	Coach, Trainer
Feedback	Extern vorgegeben	Extern modelliert	Intern modelliert

1.3.1 Der Behaviorismus

Der Behaviorismus sieht den Menschen als passives und weitgehend fremdgesteuertes Wesen. Der Wissenserwerb verläuft planmäßig und vorprogrammiert, wobei das Grundwissen des Lernenden die Basis bildet. Der Lehrer übernimmt die aktive Rolle des Wissensvermittlers, der Schüler die passive als Informationsabsorbent. Lerninhalte werden in häufig extrem kleine Lerneinheiten zerlegt. Ziel ist es komplexe Vorgänge durch eine schrittweise Annäherung zu begreifen. Richtige Antworten werden während des Lernprozesses sofort belohnt, falsche Aussagen ebenfalls unmittelbar als Fehler quittiert. Diese Art des Lernens nennt man auch Lernen durch Verstärkung oder Lernen am Erfolg. Durch die Belohnung einer richtigen Antwort, wird die positive Erfahrung verstärkt, der Lerner hat ein Erfolgserlebnis. (vgl. Thorndike 1898)

Eher biologisch betrachtet, verläuft das Lernen im Behaviorismus nach dem Reiz-Reaktions-Schema. Das Gehirn reagiert auf äußere Reize, wobei natürlich nur geeignete Reize zum gewünschten Resultat führen. Der Wissenserwerb ist hierbei die einfache Abspeicherung von Informationen. Das Gehirn kann wie eine „Black Box“ gesehen werden. Die internen Vorgänge, der Verstehensprozess, interessieren nicht, sondern nur das Resultat, die Information an sich.

(Baumgartner, Payr 1994,; Baumgartner, Payr 1994b; Haag 1995)

Der Behaviorismus ist bereits in der Vergangenheit häufig in Kritik geraten. Nachteilig ist vor allem die Vernachlässigung individueller Faktoren, komplexe menschliche Lernprozesse können nicht erfasst werden. Es liefert zwar gute Ergebnisse in der Aneignung körperlicher Fertigkeiten und wird dort auch heute noch häufig als Strategie angewendet. Jedoch liefert es keine Erklärung für entdeckendes Lernen ohne Verstärkung, wie zum Beispiel das Lernen einer Sprache während eines Auslandsaufenthaltes, ohne die Sprachschule zu benutzen. Besonders schwierig bei der Konzeption behavioristischen Lernens ist vor allem die Findung geeigneter Stimuli, um eine gewünschte Reaktion zu bekommen. Außerdem muss auch das Feedback so gewählt werden, dass positive Erfahrungen überwiegen und negative Bewertungen nicht zur Resignation des Lerners führen.

Programmtypen, die den Behaviorismus als Muster verwenden sind Drill & Practice-Programme. Sie haben einen starren Ablauf verschiedener Lernschritte und resultieren in der Erreichung vordefinierter und eindeutiger Lernziele. Auf die Präsentation kleiner Informationsmengen folgt eine Abfrage des gerade gelernten durch Kontrollfragen. Wichtig dabei ist das sofortige Feedback. Tutorielle Programme, wie Vokabel- oder Mathematik-Trainer können ebenfalls zu den behavioristischen gezählt werden.

(vgl. Will 2000)

1.3.2 Der Kognitivismus

Im Kognitivismus agiert der Lerner, im Gegensatz zum Behaviorismus, aktiv und

selbstgesteuert. Das Lernen ist hier das Verstehen und Verarbeiten von Informationen und dessen richtige Einordnung in bereits bestehendes Wissen. Kommen wir auf die „Black Box“ zurück, befinden wir uns jetzt tief in dessen Inneren und leuchten sie aus. Der Problemlösungsprozess lässt Verbindungen zwischen den Lerninhalten und individuellen Erfahrungen zu. Daraus resultierend erfolgt das Lernen nach zwei Prinzipien. Zum einen ist es ein Prozess, indem bereits vorhandenes Wissen und Erfahrungen angewandt und so erweitert bzw. angepasst werden, dass sie auf neue Aspekte der Umwelt passen. Zum anderen werden Schemata angewandt um die Umwelt zu ändern. Optimiert werden kann der Prozess durch Anleitung und individuelle Unterstützung der Lernenden durch den Lehrer, sowie die zielgerichtete Gestaltung der Lehrmaterialien.

Hierbei ist entscheidend, dass es nicht um das Finden einer richtigen Antwort geht, sondern um das Entdecken und Verstehen beliebig vieler Verfahren, die alle zur optimalen Lösung führen.

Obwohl der Kognitivismus das genaue Gegenteil des Behaviorismus bildet, gibt es auch hier Kritikpunkte, die nicht unbeleuchtet bleiben sollen. Im Kognitivismus findet eine starke Konzentration auf den individuellen Lerner statt, wodurch diese Methode nicht für die effektive Schulung von Gruppen genutzt werden kann. Auch deshalb ist der Konzeptions- und Entwicklungsaufwand enorm hoch, da es unzählige Lernverhalten zu berücksichtigen gibt. Außerdem ist das Lernverhalten schwer darstellbar, da es sich um nicht sichtbare Denkprozesse handelt. Das macht wiederum eine unterstützende Tätigkeit schwer. Kognitivistische Lehrbereiche beschränken sich auf elementare und algorithmische Domänen, wie Mathematik, Physik oder Chemie. Das Sprachenlernen fällt vorwiegend kognitivistischen Lernern meist schwer, da es bei Vokabeln lernen keine Logik gibt, die es zu verstehen gilt, sondern es sich um reine Informationsaufnahme handelt.

Intelligente Tutorielle Systeme, sowie adaptive Systeme bieten Lernern nach kognitivistischem Vorbild implementierte Verzweigungsmöglichkeiten, die das Programm und den programmgestützten Lernprozess individualisieren. Trotz der

persönlichen Ausrichtung, steht der Tutor als Beobachter und Helfer zur Seite, wenn es erforderlich ist.

(vgl. Baumgartner, Payr 1994; Haag 1995; Will 2000)

1.3.3 Der Konstruktivismus

Zur konstruktivistischen Methode gehört das problemorientierte, entdeckende Lernen. Hier ist der Lerner in besonderem Maße aktiv, denn er leitet her und konstruiert neues Wissen. Er ist im Umgang mit dem Lerngegenstand, das heißt, der Lerner entwickelt selbst Lösungswege und erlangt dadurch neues Wissen. Grundlage hierfür ist das Verständnis für ein Problem.

Im Konstruktivismus ist nicht ein Lehrer im eigentlichen Sinne gefragt, sondern in Form eines Trainers. Er sollte genügend Informationsquellen zur Verfügung stellen und den Lerner dann selbstständig agieren und probieren lassen. Der Schüler lernt durch Handeln. Ebenfalls unterstützend wirken bereits konstruierte Lösungen zu ähnlichen Problemstellungen. Lerner und Lehrer gelangen hier gemeinsam zu neuen Erkenntnissen und Erfahrungen.

Der Konstruktivismus ist schwer im Unterricht anzuwenden, da es für Lehrer und auch Lerner eine enorme Herausforderung darstellt und nur bei genügend emotionaler und geistiger Teilnahme zum Erfolg führen kann. Außerdem lässt sich nicht alles konstruieren. Manche Erkenntnis ist einfach vorhanden und bedarf keiner Herleitung.

Konstruktivistisch geprägte Programme bieten dem Lerner authentische Erlebniswelten. Probleme und Fragestellungen sind meist nicht vorgegeben, sondern müssen erst vom Lerner aus dem Gesamtkontext und unter Einbeziehung bekannter Erkenntnisse erkannt und dann behandelt werden. Praktische Beispiele sind Simulationen (*2.6.3 Simulationen*), die umfassende Situationen abbilden, jedoch die Manipulation durch den Lerner zulassen und Mirkowelten zum Entdecken eigenverantwortlichen Handelns durch den Lerner. (vgl. Baumgartner, Payr 1994; Haag 1995; Will 2000)

1.4 Der Begriff virtuell im Kontext Bildung

„Virtualität ist die Eigenschaft einer Sache, nicht in der Form zu existieren, in der sie zu existieren scheint, aber in ihrem Wesen oder ihrer Wirkung einer in dieser Form existierenden Sache zu gleichen. (...) Virtualität spezifiziert also eine gedachte oder über ihre Eigenschaften konkretisierte Entität, die zwar nicht physisch, aber doch in ihrer Funktionalität oder Wirkung vorhanden ist. Somit ist „virtuell“ nicht das Gegenteil von „real“, obwohl es fälschlicherweise oft so verwendet wird“

(Wikipedia 2009a).

Im heutigen Computer-Zeitalter hat der Begriff an Bedeutung gewonnen. Nach Welsch (2000) gibt es drei Verwendungsweisen in der elektronischen Kommunikation.

Im technischen Bereich findet man den Begriff im Zusammenhang mit der Simulation von Geräten oder Vorgängen. Virtuelles Laufwerk oder virtuelle Festplatte sind Komponenten, die als Hardware eines PCs zwar nicht physisch vorhanden sind, jedoch deren Funktionalität voll unterstützen. Dadurch kann der reale Zustand des Computers erweitert werden und somit die Nutzbarkeit erhöht.

Weiterhin unterscheidet Welsch (2000) die internetbezogene Verwendungsweise und die Verwendungsweise in „virtual reality“. Fasst man diese beiden Deutungen zusammen, geht es um realitätsnahe Simulation / Welten im Internet. Über eMail oder Chat können sich Menschen austauschen, ohne dabei physisch im Kontakt zu treten. Man trifft sich im Internet – ein häufig genutzter Ausdruck. Nur gibt es keinen realen Ort namens Internet, an dem Menschen tatsächlich aufeinander treffen, sich anfassen können. Menschen von überall auf der Welt können zur selben Zeit an einem gemeinsamen Ort sein, wobei dieser nur nachgebildet ist und nicht wirklich existiert.

Nimmt man Welsch Betrachtungsweise, könnte man schnell zu dem Schluss kommen, dass 'virtuell' gleich zu setzen mit 'nicht real' ist, dass das Virtuelle schlicht nicht vorhanden ist. Führt virtuelles Lernen zu realen Veränderungen im Verhalten und zur Erweiterung des Wissens? Stehen virtuell und real in Konkurrenz zueinander?

Mader & Stöckl (1999) sehen in virtueller Lernsoftware tatsächlich die Tendenz zur Ersetzung des Lernpartners durch Software. Jedoch bleibt fraglich, ob diese Betrachtung nicht ebenfalls eingengt ist, da der Personenbegriff sehr oberflächlich genutzt wird.

Virtuelles Lernen bedient sich der Vorteile, die die elektronische Kommunikation mit sich bringt. Lernpartner sollen nicht ersetzt werden, sondern lediglich als Alternative dienen, wenn man sich allein und selbstständig Wissen aneignen will. Soziale Aspekte im Lernprozess können so nicht bedient werden. Deshalb ist es immer sinnvoll reale Lerneinheiten lediglich durch virtuelle zu ergänzen, jedoch nicht vollständig zu ersetzen.

1.5 Zusammenfassung

Nachdem wir ein Verständnis dafür bekommen haben, dass sich Lernen und Wissensaneignung nicht ausschließlich in der Schule oder anderen Bildungseinrichtungen abspielt, sondern Teil des Alltages ist, erschließen sich völlig neue Einsatzgebiete für virtuelle Lernwelten.

Die Lernmethoden sind genauso unterschiedlich wie die Bereiche, in denen sie eingesetzt werden. Besonders gegensätzlich sind Behaviorismus und Kognitivismus. Auf der einen Seite steht der Lehrer als vollkommene Autorität, der den Lerner leitet. Hier gibt es wenig Spielraum für eigene Überlegungen. Andererseits zielt die kognitivistische Herangehensweise darauf ab, als Lerner selbst aktiv den Lernprozess zu gestalten. Beide Methoden haben autonom keinen praktischen Nutzen, da sie die Wissensaneignung stark eingrenzen würden. Eine Kombination aus Beidem findet man häufig in der Praxis, zum Beispiel im Universitätsalltag. Hier gibt es auf einer Seite die Vermittlung von reinem Faktenwissen in Form von Frontalunterricht (Behaviorismus) und auf der anderen die Erweiterung und Festigung des Gelernten durch das Selbststudium (Kognitivismus).

Der Konstruktivismus ist eine Erweiterung des Kognitivismus. Auch hier ist der Lerner

aktiv und lernt durch sein Handeln. Problemstellungen sind nicht zwangsläufig vorgegeben, sondern entstehen durch das Verstehen komplexer Situationen. Lerner erschaffen durch Anwendung vorhandenen Wissens und Herleitung von bestehenden Lösungen neue Erkenntnisse. Ein praktisches Beispiel hierfür wäre eine wissenschaftliche Arbeit, in der eine vorher gesuchte und festgelegte Fragestellung genau untersucht wird. Lerner, wie auch Lehrer, sammeln neue Erfahrungen, das genaue Ergebnis der Arbeit ist vorher ungewiss.

Virtuelles Lernen ist durchaus real. Auch hier kommt es zur Wissensaneignung. Dabei wird als Medium insbesondere das Internet genutzt. Man spricht deshalb auch von netzbasiertem Lernen.

2. Klassifizierung virtueller Welten

Um die große und ständig wachsende Zahl virtueller Welten in geeigneter Weise gruppieren zu können, sollen in diesem Kapitel nützliche Charakteristiken zur Klassifizierung erläutert werden. Die Auswahl an Eigenschaften virtueller Welten, die herangezogen werden sollen, wurden dem White Paper der Federation of American Scientists (FAS)¹ entnommen. Hier werden 5 Merkmale hervorgehoben:

- geteilter Raum / Platz,
- grafische Nutzeroberfläche,
- Direktheit / Unmittelbarkeit,
- Interaktivität und
- Persistenz.

(vgl. Kelly 2009)

Im Abschnitt Geteilter Raum / Platz wird darauf eingegangen, wie Lerninhalte innerhalb virtueller Welten veranschaulicht werden können. Weiterhin geht es um das Generieren und gemeinsame Nutzen von Inhalten. An dieser Stelle wird ein Projekt vorgestellt, das genau das ermöglicht. Über die grafische Nutzeroberfläche erfolgt jede Interaktion des Nutzers mit dem System. 2D und 3D spielen im Kapitel 2.2 *grafische Nutzeroberfläche* den Betrachtungsmittelpunkt. Direktheit / Unmittelbarkeit soll für Interaktion in virtuellen Welten mit und durch virtuelle Charaktere stehen. Hier werden Avatare vorgestellt. Wie können sie den Lerner unterstützen? Werden sie in naher Zukunft reale Lernpartner ersetzen? In diesem Zusammenhang ebenfalls wichtig werden unter 2.4 *Interaktivität* die Interaktionsmöglichkeiten mit einem System und dessen Ausprägung besprochen. Im letzten Abschnitt Persistenz wird die dauerhafte Speicherung von Daten näher erläutert, die für jedes Lernprogramm unabdinglich ist.

All das wird in diesem Kapitel bearbeitet und zu einer nützlichen Zusammenfassung

1 http://vworld.fas.org/wiki/Virtual_Worlds_Whitepaper

wichtiger Merkmale als Klassifikatoren führen. Anhand dieser Klassifikatoren wird eine Einteilung von Lernprogrammen in sechs Gruppen vorgenommen (*2.6 Anwendung der Charakteristika zur Typisierung von Lernprogrammen*). Im Anhang I befindet sich eine Übersicht mit zahlreichen virtuellen Welten, die während der Recherche gesammelt wurden. Darüber hinaus werden bereits im Text einige Programme als Beispiele genutzt, um das Gesagte zu veranschaulichen.

2.1. Geteilter Platz / Raum

Menschen treffen sich, um in der Gemeinschaft stärker zu sein und ihr Wissen durch das der anderen anzureichern. Gruppenarbeit ist ein essentieller Teil des Lernprozesses und wird auch im Berufsleben wichtiger Bestandteil bleiben. Befinden sich die Kooperationspartner jedoch an verschiedenen Orten, wird es oftmals schwierig eine effektive Zusammenarbeit zu bewährleisten. Diese beschränkt sich dann meist auf E-Mail- oder telefonischen Kontakt. Eine neue Möglichkeit der zeitnahen und direkten Zusammenarbeit können virtuelle Welten bieten. Dazu entstehen einige Fragen. Auf welche Weise können Lerninhalte in einer Lernwelt dargestellt werden? Zur Beantwortung werden die 4 Veranschaulichungsprinzipien von Prof. Dr. Stephan Schwan und Dr. Jürgen Bruder erläutert. Zum Schluss dieses Abschnittes wird ein Produkt vorgestellt, das zur Entwicklung virtueller 3D-Welten eingesetzt werden kann.

2.1.1 Veranschaulichungsprinzipien nach Schwan & Bruder (2006)

Schwan & Buder (2006) unterstützen die These, dass das Lernen in einer virtuellen Lernwelt allein durch die Verräumlichung, der Schaffung einer künstlichen Umgebung für den Lerner, und die räumliche Anordnung der Lerninhalte einen Mehrwert darstellen. Deshalb haben sie 4 Veranschaulichungsprinzipien definiert, die sich durch die Form der darin präsentierten Lerninhalte unterscheiden. Schwan & Buder (2006) definieren abbildungstreu-, schematisierende-, konkretisierende und metaphorische Veranschaulichungen.

Abbildungsgetreue Veranschaulichung

In dieser Form werden reale Sachverhalte und Situationen möglichst realitätsnah modelliert, häufig als 3D-Modelle. Diese nachgebildeten Objekte dienen Simulationen oder Trainings von realen, meist gefährlichen Situationen.



Abbildung 2.1: Techforce

Die **Abbildung 2.1** oberhalb zeigt ein Übungsszenario aus dem Lernspiel Techforce², einem Projekt des Gesamtverband der Arbeitgeberverbände der Metall- und Elektroindustrie e.V. Berlin. Es wurde für Jugendliche entwickelt, die Interesse an einer Ausbildung in der Metall- oder Elektroindustrie haben. Innerhalb des Spiels sind sie Teil eines Entwicklerteams und basteln an einem Raumschiff. Wie in der Abbildung 2.1 zu sehen, ist es zum Beispiel Aufgabe einen Schaltkreis zu basteln. Die einzelnen Bestandteile sind realistisch nachgebildet.

Abbildungsgetreue Veranschaulichungen können aber auch als Repräsentation von nicht mehr existierenden Gebäuden genutzt werden, um einen besseren Eindruck von diesen zu bekommen und Geschichte lebendig erscheinen zu lassen. In der Praxis wird diese Variante heute von Architekten angewandt, um ihren Kunden schon vorab einen Rundgang durch ihr noch nicht existierendes Heim gestatten zu können.

Diese Art der Veranschaulichung prägt sich besser ein, als der Versuch der textuellen Beschreibung. Auch gegenüber herkömmlichen Bildern in 2D überzeugen 3D-Modelle

2 <http://www.techforce.de/>

durch einen hohen Realitätsgrad. Dadurch kann sich der Lerner besser mit dem Lerngegenstand identifizieren. Die Chance, dass die virtuelle Welt vom Lerner erfolgreich auf die reale angewendet wird, ist sehr hoch.

Dieser eben als Vorteil angepreiste hohe Grad an Realismus kann sich auch negativ auf die Lernerfahrung auswirken. Hoher Realismus bedeutet detailgetreue Abbildung. Viele Details können den Lerner aber auch vom Wesentlichen ablenken.

Schematisierende Veranschaulichung

Hier wird bewußt auf Detailtreue verzichtet und der Lerngegenstand auf das Wesentliche reduziert. Angewandt wird diese Methode zur Repräsentation von sehr großen, nicht mit dem Auge erfassbaren Szenarien. Ein sehr gekanntes Beispiel, wenn auch nicht digital, ist der Globus. Er stellt eine Abstraktion der Welt dar, mit eingezeichneten Landmassen und Wasser. Es wird auf Details verzichtet, wie Flüsse oder Städte in den Ländern, da sonst die Übersichtlichkeit verloren ginge.

Schematische Abbildungen werden oft durch Texte und Grafiken ergänzt um den Sachverhalt auf mehreren Ebenen zu präsentieren und die Einprägsamkeit zu steigern. Schematisierend bedeutet so viel wie umreißend. Hierbei muss nicht unbedingt auf 3D-Elemente zurückgegriffen werden. Auch eine Zeichnung kann einen Gegenstand abstrahieren. Besonders reizvoll ist die Kombination bzw. das Ineinanderführen aus abbildungsgetreuen und schematischen Veranschaulichungen. Als Beispiel soll hier noch einmal der Globus dienen, der als interaktive Landkarte implementiert ist. Nun kann der Nutzer ein Gebiet auswählen und daraus „wächst“ dann ein 3D-Modell eines dort befindlichen Gebäudes, das der Lerner dann von allen Seiten betrachten kann.

Konkretisierende Veranschaulichung

Dieses Prinzip beschreibt die Darstellung abstrakter Sachverhalte, die für das menschliche Auge sichtbar gemacht werden, um ein besseres Verständnis für den Prozess zu bekommen. In diesen Bereich fallen physikalische oder chemische Prozesse, wie die Beschleunigung oder die Arbeit von Enzymen oder anderen Kleinstorganismen.

Man spricht auch vom Prinzip der Sinnesskalierung. Dinge, die von den menschlichen Sinnesorganen nicht wahrnehmbar sind, werden sichtbar, tastbar oder hörbar gemacht. (Winn 1993)

Um solche Veranschaulichungen verstehen und auch weiter übertragen zu können, ist es unbedingt erforderlich, dass der Lerner Vorkenntnisse über den zugrundeliegenden Sachverhalt hat.

Metaphorische Veranschaulichung

Auch bei der metaphorischen Veranschaulichung werden abstrakte Prozesse dargestellt. Anders als bei der schematischen oder konkretisierenden Darstellung geht es hier nicht darum, Sachverhalte, die so für das Auge nicht sichtbar sind durch Skalierung oder Modellbildung zu zeigen. Stattdessen werden geeignete Metapher zur Beschreibung von Prozessen genutzt.

Zum besseren Verständnis wird WhizLow, ein digitaler Agent (*2.3.1 Definition Avatar*), an dieser Stelle vorgestellt. WhizLow (Abb. 2.2) ist eine Entwicklung der North Carolina State University's IntelliMedia Initiative³.



Abbildung 2.2 WhizLow

(Quelle: http://www.dfki.de/~kipp/seminar/writeups/Mehdi_pedagogical_Agents.pdf)

³ <http://research.csc.ncsu.edu/intellimedia/index.htm>

Er bewegt sich in CPU City, einer 3D-Welt, und transportiert dort Daten als Pakete von einem zum anderen Gebäude. In der Welt repräsentiert WhizLow durch seine Handlungen Vorgänge, die innerhalb des Hauptprozessors ablaufen. Es wurde nicht ein 3D-Modell eines Prozessors gewählt, sondern das Methaper einer Stadt. Die verschiedenen wichtigen Komponenten des Hauptprozessors bilden die Gebäude dieser Stadt und WhizLow symbolisiert durch das Herumtragen von Datenpaketen von einem zum anderen Gebäude die tatsächlich stattfindenden Datenübertragungen innerhalb der CPU.

Entscheidend für metaphorische Veranschaulichungen ist die Wahl geeigneter Metapher. Anderenfalls könnte der Lerner falsche Assoziationen bilden und so falsches Wissen produzieren.

2.1.2 Sharable Content – Inhalte gemeinsam nutzen

Lernsoftware bietet die Möglichkeit, Inhalte zu gruppieren und einer breiten Masse von Lernern zugänglich zu machen. Insbesondere durch online-Anwendungen müssen sich die Lerner nicht länger im selben Raum befinden, um dieselben Inhalte vermittelt zu bekommen. Jedoch sind die Inhalte bisheriger Lernanwendungen fest vordefiniert und können von den Lernenden selbst nicht geändert, sondern nur genutzt werden.

Eine neue Möglichkeit des Austausches bzw. der gemeinsamen Nutzung von Inhalten bietet Project Wonderland⁴. Hier können die Anwender nicht nur auf vorgefertigte Inhalte und Übungen zurückgreifen, sondern sie können ihre Desktop-Anwendungen in die Welt einbinden und somit Anderen die Mitarbeit an eigenen oder gemeinsamen Projekten oder Übungen gestatten. Project Wonderland bietet interessierten Nutzern einen Baukasten mit Werkzeugen zum Erstellen von 3D-Welten. Diese Welten eignen sich besonders gut für die Arbeit in international tätigen Firmen, aber auch für die Abhaltung von Lehrveranstaltungen im virtuellen Klassenzimmer. Sie ermöglichen den geografisch weit gestreuten Mitarbeitern bzw. Lernern das Arbeiten von einem belie-

4 <https://lg3d-wonderland.dev.java.net>

igen Platz aus. Trotzdem haben diese das Gefühl, in einem Raum mit den Anderen zu sein. Die Nutzer können sich austauschen und ihre Arbeit für die Anderen zugänglich machen bzw. gemeinsam Dokumente nutzen.

Die 3D-Komponenten für die Welt, können derzeit nur mithilfe externer Programme, wie Maya⁵ erstellt und dann als x3d⁶-Dateien in die Welt importiert werden. In Zukunft soll die Entwicklung von Content auch innerhalb der Welt möglich sein. Externe Tools werden auch dann weiter unterstützt, da sie dem Autor einen viel größeren Handlungsumfang bieten können als inworld – Anwendungen.

Project Wonderland basiert auf folgenden Open Source Projekten, die die Haupttechnologie der Plattform bilden und alle in Java geschrieben sind:

- Project Darkstar⁷ - ein Game Server, der sich durch Skalierbarkeit und Beständigkeit auszeichnet
- jVoiceBridge⁸ – für Audio in Stereoqualität mit Dämpfung bei Entfernung,
- Java 3D⁹ – ein Szenengraph, auf dem der Szenenmanager aufgebaut ist und der alle Objekte der Welt verwaltet und
- Project Looking Glass¹⁰ – einem 3D Szenenmanager.

Derzeit ist es möglich X11-Anwendungen in die Welt einzubinden, um sie dort gemeinsam zu nutzen. Diese Anwendungen laufen auf dem Server und können vom Client aus gesteuert werden, wobei Server und Client nicht auf dem selben Rechner laufen müssen. Der Client erkennt Maus- und Tastatureingaben, die er dann an den Server übermittelt. Die X11 Anwendungen erscheinen als Bildschirm in der virtuellen Welt und können von den verschiedenen Nutzern von dessen Client aus gesteuert werden. Wenn ein Nutzer in der virtuellen Welt den Rahmen des Fensters anklickt, färbt sich dieser grün. Das

5 <http://www.autodesk.de/adsk/servlet/pc/index?id=13743043&siteID=403786>

6 x3d ist ein Dateiformat zur Speicherung von dreidimensionalen Inhalten

7 <http://www.projectdarkstar.com/>

8 <https://jvoicebridge.dev.java.net/>

9 <https://java3d.dev.java.net/>

10 <https://lg3d.dev.java.net/>

symbolisiert ihm, dass er nun auf die Anwendung zugreifen kann. In diesem Zusammenhang hat der Client eine Anfrage an den Server gesendet. Der Server gibt die Anwendung für den anfragenden Client frei und informiert den Nutzer über dessen erlangte Handlungsfreiheit, indem er das Einfärben des Bildschirmrahmens veranlasst. Gleichzeitig werden alle anderen Clients darüber informiert, dass sie momentan nur zuschauen können. Wählt ein anderer Nutzer nun ebenfalls das Fenster aus, färbt sich der Rahmen für den eben aktiven Nutzer rot und er hat keinen Zugriff mehr. Der Client des neuen Nutzer schickt mit dem Klick eine Anfrage an den Server. Dieser gibt die Anwendung für den neuen Nutzer frei und sendet dem vorherigen Client eine Nachricht, dass dieser nicht mehr aktiv ist.

Darüber hinaus stehen einige Anwendungen in der Welt zur Verfügung, auf die mehrere Nutzer gleichzeitig zugreifen können, wie der PDF-Viewer zum Anschauen und Navigieren innerhalb von pdf-Dateien oder das SVC Whiteboard zum Zeichnen.

Project Wonderland ist zu 100 Prozent Open Source und deshalb für jederman frei erhältlich. Es wird bereits erfolgreich von Unternehmen zur internen Kommunikation, aber auch zur Zusammenarbeit mit Partnern oder Kunden genutzt. MPK20 ist beispielsweise der virtuelle Arbeitsplatz von Sun Microsystems¹¹, an dem sich Mitarbeiter treffen um sich auszutauschen oder ihre Arbeit zu präsentieren. Außerdem werden Meetings innerhalb der virtuellen Welt abgehalten.

Eine weitere Open Source Alternative zur Entwicklung von virtuellen 3D-Welten, die ähnliche Möglichkeiten zur gemeinsamen Nutzung von Inhalten bietet, Croquet Community¹², soll an dieser Stelle nur genannt werden. Ebenso gibt es kommerzielle Produkte wie Forterra's OLIVE Plattform¹³, die vorallem vom US Militär eingesetzt wird.

11 <http://research.sun.com/projects/mc/mpk20.html>

12 <http://www.opencroquet.org>

13 <http://www.forterrainc.com>

2.2 grafische Nutzeroberfläche

Die Gestaltung der Nutzeroberfläche ist entscheidend für die Bedienbarkeit eines Systems. Das lässt sich natürlich auch auf virtuelle Lernwelten übertragen. Um den Nutzer einen leichten Einstieg in die Welt und somit direkten Zugang zu den Lerninhalten garantieren zu können, sollte das Interface ansprechend und intuitiv gestaltet sein.

Früher waren Spieloberflächen und auch die allerersten eLearning-Anwendungen in 1D. Eindimensionale Anwendungen sind Konsolen, über die das Programm durch Texteingaben gesteuert wird. Es gab keine Grafiken oder Videos. Die Navigation und Interaktion erfolgte ausschließlich textbasiert. Die Bezeichnung 1D ist mathematisch nicht ganz korrekt, da es einen Punkt beschreibt, Text jedoch bereits eine Fläche und deshalb zweidimensional ist. In der Computerbranche hat er sich jedoch etabliert und beschreibt nach wie vor die rein textbasiert-gesteuerten Programme.

Heutzutage sind nur noch sehr selten reine 1D-Anwendungen zu finden. Vielmehr werden sie als Komponente zur Kommunikation in vielen 2D- und 3D-Anwendungen., z.B für Chatbots (2.3.3. *Eingesetzte Technologien*), eingesetzt.

2.2.1 2D und 2,5D

Die meisten Nutzeroberflächen in Lernanwendungen sind 2D oder 2,5D. Im zweidimensionalen Raum lässt sich jeder Punkt durch 2 Parameter beschreiben. Eine virtuelle Welt in 2D ist also stets eine flache Abbildung, ohne Tiefen. 2,5D beschreibt in diesem Zusammenhang die isometrische Variante. Dabei wird die Szene aus einem anderen Winkel betrachtet, beispielsweise von schräg oben, jedoch ohne die perspektivische Verzerrung darzustellen. Durch den richtigen Einsatz von Schatten und einer perspektivisch korrekten Darstellung können 3D-Umgebungen in 2D simuliert werden.

Adobe Flash¹⁴ ist derzeit das am häufigsten genutzte Autorentool zur Erstellung multimedialer Anwendungen, wie Lernprogramme, Spiele oder Animationen in 2D, die im Web veröffentlicht werden. Grund dafür sind neben den gebotenen Möglichkeiten, die die Entwicklungsumgebung mit sich bringt, die weite Verbreitung des Flash-Plug-in¹⁵ für Webbrowser. Wichtigster Konkurrent von Adobe Flash ist Microsoft Silverstar¹⁶. Zwar ist das Browser-Plug-in momentan noch nicht so verbreitet wie der Flash-Player, jedoch ist das nur eine Frage der Zeit, da es aus der Microsoft-Familie stammt und dieses das dominierende Betriebssystem für Endanwender ist. Silverlight bietet dem Entwickler ähnliche Möglichkeiten wie Flash, wobei es durch qualitativ besonders hochwertiges Videostreaming punkten kann.

Die Anwendungen können in ihrer Gestaltung stark variieren. Bei Programmen für Kinder und Jugendliche wird häufig der Comic Stil angewandt, wie das Beispiel in **Abbildung 2.3** zeigt.



Abbildung 2.3: Babus Hold¹⁷

Hierbei werden alle Objekte stark mit schwarz umrandet und die Flächen meist ohne Verläufe, sondern mit reinen Farben, gefüllt. Andere Anwendungen nutzen Illustrationen oder Videosequenzen bzw. Fotos zur Gestaltung der virtuellen Lernum-

14 <http://www.adobe.com/de/products/flash/>

15 <http://www.adobe.com/de/products/flashplayer/>

16 <http://www.silverlight.net/>

17 <http://www.babushold.dk/>

gebung. Fotorealistische Abbildungen einer virtuellen Welt werden vorwiegend für die Aus- und Weiterbildung von Erwachsenen genutzt, Illustrationen finden in beiden Alterstufen gleichermaßen Einsatz.

2.2.2 3D

Um virtuelle Welten realistischer gestalten zu können, wird zunehmend auf 3D zurückgegriffen. Im Vergleich zu 2D-Umgebungen stellen 3D-Welten vorallem in benötigter Rechenleistung und Speicher wesentlich höhere Anforderungen. Insbesondere im Web waren 3D-Objekte bis vor kurzem noch selten verwendbar. Zwar gab es schon 1995 erste Ansätze von der Firma Silicon Graphics (SGI)¹⁸, jedoch waren diese aufgrund der damals handelsüblichen leistungsschwächeren PCs bzw. der hohen Kosten für die benötigte Hardware nicht für die Allgemeinheit zugänglich. SGI entwickelte ein Inventor-Format, was die Basis für das spätere VRML-Format bildete. Mithilfe des VRML-Format können 3D-Objekte in HTML-Seiten geladen, dargestellt und sogar verlinkt werden. Heute sind Rechenleistung und Bandbreite so gut, was die Darstellung von 3D-Objekten im Web in Sachen Wartezeit und Kosten ermöglicht. (Krause 2008)

3D-Lernumgebungen eignen sich für Lerner jeden Alters. Jedoch ist besonders bei Welten für jüngere Lerner darauf zu achten, dass die Welt so gestaltet ist, dass der Lerner durch zahlreiche Details und Interaktionsmöglichkeiten nicht vom eigentlichen Lerngegenstand abgelenkt wird.

Papervision3D¹⁹ ist eine 3D-Render-Engine, die eine Erweiterung des Flash-Plug-in darstellt. Damit können 3D-Animationen erstellt und leicht im Internet zugänglich gemacht werden. Eine Alternative ist Shockwave3D²⁰, eine 3D-Engine des Autoren-systemes Adobe Director²¹.

18 <http://www.sgi.com/>

19 blog.papervision3d.org/

20 <http://www.adobe.com/products/shockwaveplayer/>

21 <http://www.adobe.com/products/director/>

2.2.3 Die Navigation in 2D- und 3D-Welten

Die Navigation innerhalb einer 2D-, 2.5D- oder 3D-Welten kann über zwei Wege erfolgen, point-and-click oder Cursor-Steuerung. Für die Cursor-Steuerung werden die Pfeiltasten auf der Tastatur verwendet. Hingegen beschreibt point-and-click das Auswählen und Anklicken der Zielposition mit der Maus. Durch das Drücken der linken Maustaste gelangt man dann von der aktuellen zur neuen Position. Beide Varianten können sehr intuitiv genutzt werden, wenn klar ist, wo in der Umgebung es was zu entdecken gibt. Fällt die Orientierung und Navigation innerhalb der Welt schwer, führt das zu einer Verzögerung oder Ablenkung vom Lernprozess.

Man kann nicht grundsätzlich sagen, dass 2D- oder 3D-Programme einfacher zu bedienen sind da dies stark abhängig von der Organisation der Inhalte und Navigationselemente ist und weniger von der grafischen Gestaltung. Auf jeden Fall kann man sagen, dass eine optisch ansprechende Oberfläche so manchen Mangel in der Bedienfreundlichkeit vergessen lässt.

2.3 Direktheit / Unmittelbarkeit

Avatare werden immer häufiger als personelle Präsentation von Nutzern eingesetzt und bilden somit die Hauptschnittstelle zwischen Lerner und dem System. Die Aufgaben, die sie dabei übernehmen, die genutzten Technologien zur Steuerung und der Mehrwert, den ihr Einsatz mit sich bringt ist Inhalt dieses Abschnittes.

2.3.1 Definition Avatar

Ein digitaler Agent oder Avatar ist im weitesten Sinne Software, die einen Nutzer repräsentiert und in dessen Namen agiert, wobei das Programm weitestgehend autonom arbeitet. Dabei folgt die Software dem Simulationsmodell eines Menschen, z.B. durch die Digitalisierung eines Körpers, der sich fortbewegen und ausdrücken kann, durch

Sprache oder auch Gesten. Das Skript, dem die Bewegungen eines Avatars folgen, gibt Verhaltenssets vor, die sich bei hohem Interaktionsgrad (2.4 Interaktivität) dynamisch anpassen. In diesem Fall ist der animierte Charakter in ein AI-Backend, mit einem Stimmungs- und Verhaltenssystem zur Simulation menschlicher Emotionen und Aktionen, gebunden. Mehr zu dieser Funktionsweise unter 2.3.3 *Eingesetzte Technologien*.

„Wenn man sie braucht, sind sie da. Ihren Rat schätzt man, ihre Strenge braucht man nicht zu fürchten. Wenn man will, klickt man sie einfach weg.“ (Bendel 2003)

Wie das Zitat vermuten lässt, sind Avatare die perfekten virtuellen Hilfskräfte. In dieser Rolle kann der Avatar meist die selben Aufgaben ausführen wie eine reale Person. Einschränkungen gibt es aus Technologiegründen vor allem in der gesprochenen Sprache. Neben dieser Einsatzmöglichkeit werden Avatare auch zur Repräsentation der eigenen Person genutzt. Hierbei wird die Kamera häufig im Rücken oder seitlich des Agenten positioniert. Auch andere anwesende reale Personen, können in der virtuellen Lernumgebung durch einen Avatar verkörpert werden. Lehrer oder Mitlerner werden so repräsentiert. Eine dritte Möglichkeit ist der Einsatz von digitalen Agenten als Ersatz für Teammitglieder, die nicht anwesend sein können. In diesem Fall werden sie nicht von einem Lerner, sondern vollständig vom System gesteuert.

2.3.2 Aufgaben eines Avatars

So unterschiedlich die Rollen sind, die ein Avatar einnehmen kann, so verschieden sind auch die Aufgaben, die er in der jeweiligen Rolle übernimmt.

Avatar als Repräsentation der eigenen Person

Die Wirklichkeit wird als Modell abgebildet und der Lerner erhält darin die Möglichkeit reale physikalische Arbeitsprozesse zu testen. Dabei nimmt der Avatar meist die Rolle der eigenen Person ein, um damit Unmittelbarkeit herzustellen. Hier können Fehler gemacht werden ohne, dass tatsächlich Schaden entsteht. Das heißt, der Lerner kann

schwierige Situationen proben, ohne selbst gefährdet zu werden oder andere durch Fehlentscheidungen zu gefährden.

Avatar als Navigationshilfe

Diese Variante wird vor allem für komplexe Welten genutzt. Der Avatar kann den Lerner herumführen und ihm so die Orientierung erleichtern. An dieser Stelle soll Steve vorgestellt werden.

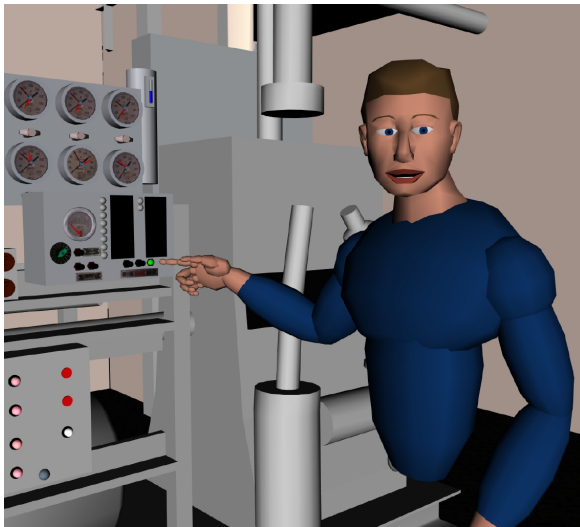


Abbildung 2.4: Steve

Steve (Abb. 2.4) ist ein digitaler Agent in einer U-Boot Simulation, die vom USC Information Sciences Institute's Center for Advanced Research in Technology for Education (CARTE)²² entwickelt wurde und zu Trainingszwecken bei der US Navy eingesetzt wird. Steve führt die Lerner durch das virtuelle U-Boot, zeigt auf wichtige Elemente und erläutert diese. Er hat ein räumliches Modell des U-Bootes implementiert, sodass er zu jeder Zeit den kürzesten Weg zwischen zwei relevanten Punkten berechnen kann.

An dieser Stelle sind ergänzend die Oberflächen-Helfer zu nennen, die dem Nutzer die Arbeit mit der Nutzeroberfläche erleichtern sollen. Das wohl bekannteste Beispiel hierfür ist der Microsoft Paper Clip. Sie erscheinen teilweise unaufgefordert und ziehen den Fokus auf sich. Der Nutzer kann nicht einfach weiter arbeiten, ohne sie zu beachten.

²² <http://www.isi.edu/isd/carte/carte-home.htm>

Dabei verhalten sie sich wie eine Dialogbox. Der Nutzer kann Fragen eingeben und der Agent antwortet mit einer Auswahl an scheinbar passenden Artikeln aus dem System. Die Erfolgsquote ist abhängig vom im System definierten Fragenkatalog.

Avatare als „Aufmerksamkeits-Führer“

Blicke und Gesten sind die natürlichste Methode um Aussagen zu verstärken und die Aufmerksamkeit der Zuhörer und Zuschauer zu lenken. In alternativen Systemen werden Rahmen oder Pfeile und kräftige Farben genutzt um den Fokus des Nutzers auf wichtige Bereiche eines Bildes zu lenken oder um wichtige Aussagen zu unterstreichen. In dieser Rolle des Tutor gibt der Avatar den Stoffleitfaden vor und begleitet den Lerner auf seinem Weg zur Lösung.

Steve zum Beispiel nutzt seine Hand, um damit auf die wichtigen Instrumente zu zeigen, schaut auf das Instrument, wenn er oder ein anderer Nutzer es manipuliert und wendet sich seinem Gesprächspartner durch Drehen des Kopfes zu.

Ein weiteres Beispiel ist PPP Persona, ein objektorientierter Präsentationsagent, der vom Deutschen Forschungszentrum für künstliche Intelligenz (DFKI)²³ entwickelt wurde. Dabei empfängt der PPP-Persona-Server Präsentationsaufgaben von anderen Anwendungen und führt diese dann selbstständig aus. Der digitale Agent kann dabei eine gezeichnete Figur oder eine Videosequenz sein. PPP Persona zeigt mittels eines Pointers auf Objekte auf dem Bildschirm, wie die **Abbildung 2.5** verdeutlicht.

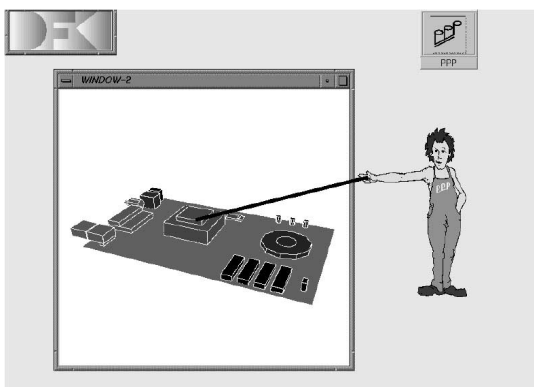


Abbildung 2.5: PPP Persona

23 <http://www.dfki.de/web>

Avatare geben nonverbales Feedback und nutzen Gesprächssignale

Eine der wichtigsten Rollen eines Trainers oder Tutors ist es dem Lerner Rückkopplung zu seinen Aktionen zu geben. Nonverbales Feedback wirkt gegenüber der üblichen verbalen Form weniger aufdringlich. Ein Kopfnicken gibt dem Lerner eine Bestätigung, ohne ihn in seinem Handeln zu unterbrechen, wogegen ein Ausdruck des Erstaunens oder der Verwirrung ein deutliches Anzeichen darauf gibt, dass der Lerner im Begriff ist einen Fehler zu begehen und ihn so zwei mal über eine Aktion nachdenken lässt.

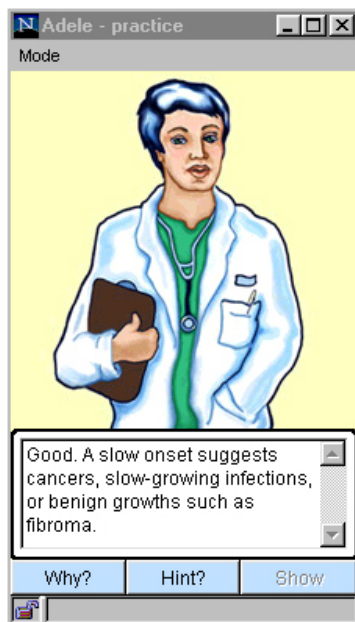


Abbildung 2.6: Adele²⁴

Adele (Abb. 2.6) ist ein weiterer Avatar, der vom CARTE entwickelt wurde. Es handelt sich um eine einfache Desktopanwendung, die den Lerner in den medizinischen Stoff einführen soll. Durch ein Nicken oder Lächeln äußert Adele ihr Einverständnis zu den Aktionen des Nutzers. Bei erfolgreichem Abschluss einer Aufgabe zeigt sie sich angenehm überrascht. Unterstützt wird das nonverbale Feedback durch synthetische Sprache.

Auch Steve äußert seine „Einstellung“ gegenüber Nutzereingaben durch zustimmende oder ablehnende Gesichtsausdrücke.

²⁴ <http://www.isi.edu/isd/ADE/ade.html>

Insbesondere Gesprächssignale haben nicht nur für das Feedback eine wichtige Funktion, sondern unterstützen die Kommunikation an sich. Mimik und Gestik hilft beim Regulieren des Ablaufes einer Unterhaltung. Durch Augenkontakt zwischen Gesprächspartnern weiß man zu jeder Zeit, wer an wen gerichtet spricht, wodurch die Kommunikation persönlicher erscheint. Kurze Augenbewegungen oder ein Zwinkern lassen die Unterhaltung natürlicher wirken.

Natürlich ist grundsätzlich Kommunikation auch ohne solche Gesprächssignale möglich, wie zum Beispiel bei einem Telefonat. Jedoch kann es auch leichter zu Missverständnissen kommen.

Durch Gesichtsausdrücke und Gesten verstärkt der Avatar seine emotionale Wirkung. Emotionstypen sind neben Freude und Glücklich sein auch Angst, Schadensfreude und Neid, wobei ausschließlich positives emotionales Feedback den Lerner motiviert und vorranbringt. Daher ist ein Teil der Emotionen in virtuellen Lernwelten unbrauchbar. Das emotionale Verhalten muss zur Laufzeit generiert werden und steht in direktem Bezug zum Verhalten und den Aktionen des Lerner.

Avatare als Virtuelle Teammitglieder

Gruppenfähigkeit ist in allen Branchen ein wichtiges Kriterium. In manchen Berufen ist sie sogar unerlässlich, da die Aufgaben nur im Team zu bewältigen sind. Um die Zusammenarbeit effektiv zu trainieren eignet sich der Einsatz digitaler Agenten in virtuellen Welten. Jedes Mitglied kann seine eigenen Handlungen üben und das Zusammenwirken mit den anderen Teammitgliedern koordinieren. Die virtuelle Realität wird zu einem Übungsplatz, auf dem Lerner von unterschiedlichen Orten aus reale Situationen proben können. Dabei kann der Avatar verschiedene Rollen einnehmen. Zum einen kann er als Lehrer eingesetzt werden, um das Team zu leiten. In dieser Position motiviert ein digitaler Agent durch bloße Anwesenheit den Lerner zu besserer Leistung bzw. fördert dessen Disziplin. Oder er ersetzt fehlende Mitglieder in einer Gruppe, die nicht am Training teilnehmen können. Eine weitere Aufgabe ist die Vermittlung zwischen den Gruppenmitgliedern, wobei der Avatar Transaktionsprozesse, z.B. zum Austausch von Dokumenten oder Objekten, vorbereitet und durchführt. Dem

Agent kann aber auch die Rolle eines „Troublemakers“ einnehmen, der zeitweise falsche Informationen verteilt, um die Leistung und Willenskraft der Lerner zu testen. Egal welche Position ein Avatar einnimmt, immer ist deutlich, dass Lernen nicht nur Wissensaneignung ist, sondern auch ein sozialer Prozess.

Glaubhaftigkeit

Die Glaubhaftigkeit digitaler Agenten ist im wesentlichen von zwei Eigenschaften abhängig, dem visuellen Erscheinungsbild und dem System, das das Verhalten kontrolliert und steuert. Wirken Avatare roboterhaft, verlieren sie enorm bis komplett an Glaubhaftigkeit. Zeigt ein digitaler Agent ein unaufdringliches natürliches Verhalten, gewinnt er an Glaubhaftigkeit. Generell kann gesagt werden, dass Bewegungsabläufe, die die Interaktion zwischen Menschen auszeichnet, auch zur Steigerung der Glaubhaftigkeit von Avataren beiträgt.

Es gibt verschiedene Techniken um einen Avatar glaubwürdig erscheinen zu lassen. Zum einen sollten Gesichtsausdrücke reguliert eingesetzt werden, um den Nutzer nicht durch zuviel visuellen Einfluss abzulenken. Um über eine längere Zeit mit dem Lerner interagieren zu können, sind sehr komplexe und abwechslungsreiche Verhaltensmuster nötig. Deshalb ist es oftmals kritisch, dass vom System passende Reaktionen unmittelbar veranlasst werden können. Glaubhaftigkeit setzt Natürlichkeit voraus. Natürlichkeit kann durch den Einsatz von Verhaltensmustern erreicht werden, die Lücken zwischen Aktivitäten füllen, wie beispielsweise das Tippen mit dem Fuß oder Atmen.

2.3.3. Eingesetzte Technologien

Hinter einem intelligenten Avatar liegt ein komplexer Verhaltens-Controller, der aus einem Set von Verhaltensmuster, Nutzereingaben und dem Kontext der hinterlegten Repräsentation der Welt die Bewegungen des Avatar generiert. Anders als bei einem Film, der vorher produziert wurde, muss bei Avataren in Echtzeit gerendert und somit die Bewegungen erst zur Laufzeit errechnet werden.

Die Kommunikation mit Avataren erfolgt meist über Chatbots. Der Grund für diese Art der Kommunikation ist, dass die Spracherkennung technisch noch nicht ausgereift ist und deshalb die Kommunikation durch „Nichtverstehen“ von Akzenten o.ä. enorm erschwert werden würde.

Chatbots sind Konsolen, über die der Austausch in schriftlicher Form mit einem Avatar erfolgt. Chatbots bilden aus den Eingaben Muster und vergleichen diese mit Antwortmustern, die in einer Datenbank vordefiniert sind, auf Übereinstimmung. Mit der Eliza-Software²⁵ gibt es eine neue Möglichkeit zur Generierung von eigenen Antworten, wobei die Nutzereingaben einfach in eine Gegenfrage umgewandelt werden.

Problem an der Chatbot-Variante ist die fehlende persönliche Note. Außerdem gibt es kein zusammenhängendes Gespräch, da dieses nur aus einer Folge von Musterabgleichen generiert wird. Manchen Avataren wird die Fähigkeit des Smalltalk bis zu einem bestimmten Punkt implementiert. Das wirkt sich meist jedoch negativ aus, da die Kommunikation dadurch stark gekünstelt wirkt und wiederum an Glaubhaftigkeit verliert. Außerdem scheint persönliche Kommunikation die Aufgaben eines Avatars zu sprengen und wird deshalb oft als lästig empfunden.

Hat die Datenbanksuche eine passende Antwort oder Frage gefunden, muss die Sprache noch generiert werden, falls das Avataren-System über einen Sprach-Synthesizer verfügt. Dieser wandelt die gewünschte Nachricht des Avatar an den Lerner in einen gesprochenen Satz um und gibt es wieder. Die Lippen bewegen sich synchron, wenn der Agent mit genügend Informationen über Beginn und Ende der Rede, das Timing der Betonung und zur Koordination nonverbaler Signale vom System versorgt wird. Nur wenn alle Faktoren in ausreichendem Maß vorhanden sind, kann die Sprache eines Avatar gewissermaßen natürlich wirken.

Wegen der Möglichkeit der Kommunikation über den Avatar, dessen Wahrnehmung von der virtuellen Lernumgebung und seine Fähigkeit diese zu beeinflussen, bildet er für

²⁵ ELIZA ist ein Computerprogramm zur Kommunikation zwischen Mensch und Computer über die natürlich Sprache. Es wurde 1966 von Josef Weizenbaum entwickelt. (Wikipedia 2009b)

den Nutzer eine wichtige Schnittstelle zur virtuellen Lernwelt. Digitale Agenten speichern den aktuellen Problemstatus und die Schritte, die schon zu dessen Lösung eingesetzt wurden. Außerdem bekommt der Avatar bei jeder Veränderung in seiner Umgebung eine Nachricht, um in Echtzeit darauf reagieren zu können.

Steve zum Beispiel bekommt eine Nachricht von der Software der virtuellen Welt, die ihm mitteilt, welche Objekte der Umgebung sind im Blickwinkel des Lerners befinden. Sollte der Lerner sein Augenmerk auf ein anderes Objekt verlagern, reagiert Steve sofort, indem er seine Rede unterbricht und so lange wartet, bis der Schüler seine Augen wieder auf ihn gerichtet hat.

Zusätzlich benötigt der Agent räumliche Informationen, um sich in seiner Umgebung bewegen zu können. Wie schon zu einem früheren Zeitpunkt erwähnt, werden diese benötigt, um zum einen nicht mit anderen Objekten auf ihrem Weg zu kollidieren und zum anderen um den kürzesten Weg zwischen zwei Zielen zu finden. Um dieses Problem zu lösen kann im System des Avatar ein Modell der Welt implementiert werden, z.B. in Form eines Adjazenzgraphen, wie es bei Steve der Fall ist. Jedes ansteuerbare Element der Welt wird als Punkt eingetragen. Linien zwischen den Punkten symbolisieren einen kollisionsfreien Weg. Zur Analyse, welches nun der kürzeste Weg zwischen zwei Knoten ist, wird der Dijkstra-Algorithmus angewandt.

Bei WhizLow, den wir schon unter *2.1.1 Veranschaulichungsprinzipien nach Schwan & Buder* kennengelernt haben, wird mittels des A*-Algorithmus eine 2D-Repräsentation der 3D-Welt errechnet und abgespeichert. Die Bewegungen zwischen den Kontrollpunkten müssen von der Render-Engine interpoliert werden, sonst würden die Bewegungen „eckig“ und unnatürlich wirken, da der Agent immer auf geradem Weg von einem zum anderen Knoten gehen würde.

Zusammenfassend hat ein digitaler Agent drei wesentliche motorische Fähigkeiten. Die Sprache, und damit das Versenden von Zeichenketten, die ausgegeben werden sollen an einen Sprach-Umwandler, bildet die erste wichtige Fähigkeit. Außerdem hat er die Kontrolle über seinen Körper. Dazu gehört das Abspielen (vorgefertigter) Ani-

mationsclips für den ganzen Körper, sowie die Kontrolle einzelner Bewegungen, wie Blicke, Gesten oder Gesichtsausdrücke. Weiterhin kann er Objekte seiner Umwelt manipulieren und dadurch die Umgebung verändern. Um eben genanntes Verhalten zu erreichen, werden Avatare auf zwei wesentliche Arten implementiert. Zum einen kann das Verhalten vorher definiert und in Blocks gespeichert sein, oder es wird zur Laufzeit dynamisch generiert. Beides soll im Anschluss kurz erläutert werden.

Gespeichertes Verhalten

Dem Agenten wird vorab eine Bibliothek mit Verhaltensfragmenten implementiert. Es gibt drei wesentliche Verhaltenstypen:

- die visuellen Segmente, die alle Bewegungen des Avatar beschreiben,
- Audioclips mit allen Sprachausgaben, ebenfalls vorab fest definiert und
- Fragmente mit Musik, die die Stimmung unterzeichnen soll, d.h Hintergrundmusik und Geräusche.

Die visuellen Segmente sind in der Regel einfache Bitmaps, die die Agenten in all ihren Posen zeigen, sowie die Animationsschritte für die Bewegungen von einer zur nächsten Pose. Es können aber auch Videosequenzen sein oder Fotos realer Personen.

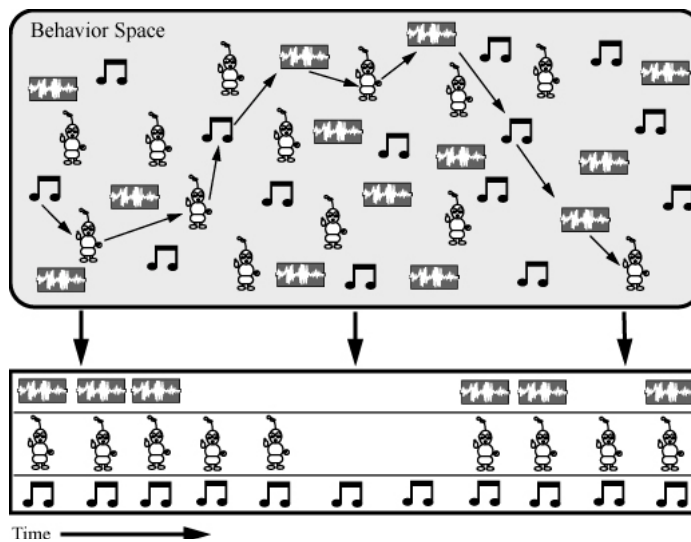


Abbildung 2.7 Verhaltensgenerierung Herman the Bug (Quelle: Lester et al. 2000)

Abbildung 2.7 zeigt die Verhaltenssegmente von Herman the Bug und dessen Anordnung zur Laufzeit. Herman the Bug ist ein Käfer, der den Lerner durch eine

Lernumgebung führt und alles über die dort dargestellten Pflanzen erklärt. Herman ist, wie auch WhizLow das Werk der North Carolina State University. In der Grafik dargestellt sind Bild-, Sprach- und Musiksegmente. Die Pfeile bestimmen ihre Anordnung zur Laufzeit. Die Verhaltensbibliothek von Herman the Bug umfasst 30 Animationen und 160 Audioclips, außerdem eine Soundbibliothek, aus der zur Laufzeit Musik gemixt wird, um Nutzeraktionen zu unterstreichen.

Die größte Herausforderung ist die Generierung flüssig erscheinender Bewegungen aus den vordefinierten Bitmaps. Um das zu erreichen eignet sich das „visual bookending“, wobei der Start- und Endframe einer jeden Animation identisch ist. Dadurch können Verhaltensfragmente in jeder beliebigen Reihenfolge angeordnet werden, was eine enorme Erleichterung bei der Generierung von Bewegungen bedeutet.

Adele beispielsweise erreicht ihre verschiedenen Posen durch Animation vorgefertigter Bitmaps. Bei PPP Persona wird eine Kombination aus statischen Bibliotheken und der dynamischen Generierung des Verhaltens angewendet. Die Bitmaps zeigen die Posen des Präsentationsagenten, wohingegen der Pointer dynamisch erzeugt wird, um auf spezielle Bereiche und Einheiten zu zeigen. Die Aufgabe des Systemes ist nicht nur die Generierung, sondern auch die Kombination beider Elemente.

Die eben beschriebene Technik der Speicherung vordefinierten Verhaltens bringt qualitativ sehr hochwertige Bewegungsanimationen hervor, da die Bilder mit viel Sorgfalt vorab produziert werden. Dadurch haben ausgebildete Animatoren sehr gute Kontrolle über die Bewegungsabläufe. Dieser Animationsprozess ist nicht nur zeitintensiv, sondern bringt auch geringe Flexibilität der Anwendung mit sich, da das gesamte Verhalten vorprogrammiert ist. Ein weiterer Minuspunkt für diese Technik ist, dass der Nutzer nur eine begrenzte Ansicht der Objekte in der Welt bekommt. Deshalb wird es eher für Agenten in 2D-Welten eingesetzt.

Dynamisch generiertes Verhalten

Solches Verhalten ist vor allem Avataren in 3D-Welten implementiert, wie z.B. WhizLow und Steve. Das grafische Modell der Agenten besteht aus einzelnen

beweglichen Teilen. Es wird ein Algorithmus programmiert, der die aktuelle Pose spezifiziert und die Bewegung des Körpers von einer in eine neue Pose generiert. Die Schwierigkeit dabei ist es, dass mehrere Körperteile synchron bewegt werden müssen, bzw. die Bewegung sich auf die einzelnen Körperteile sinnvoll fortsetzen muss.

Ähnlich funktioniert die Generierung der Sprachausgabe. Wie schon oben beschrieben werden die Texte an einen Synthesizer übergeben, der daraus die Sprache erzeugt. Sprachumwandler können individuelle Phoneme²⁶ und kontextbezogene Betonung (traurig, wütend, ...) unterstützen.

Der Vorteil dieser Technik liegt klar auf der Hand, die Anwendung erhält ein hohes Maß an Flexibilität, da der Agent zur Laufzeit situationsbezogen sein Verhalten anpassen kann, und nicht auf vorgefertigte Verhaltensfragmente zurückgreifen muss. Der Nachteil ist ebenso offensichtlich. Die Bewegungen und die Sprache wirken fast immer sehr künstlich und „eckig“. Die Animation und Gestalt der digitalen Agenten ist qualitativ minderwertiger als bei vorgefertigten Zeichnungen bzw. Grafiken.

Einsatzbereich

Eine wünschenswerte Plattform zur Nutzung wäre das Internet. Dadurch könnte eine große Menge von webbasierten Inhalten bedient werden. Das größte technische Problem ist dabei auch heute noch eine langsame Antwortzeit, wenn dem Nutzer keine optimale Bandbreite zur Verfügung steht. Wenn das Verhalten dynamisch zur Laufzeit generiert wird, kann es dadurch zu stockenden Bewegungen kommen, was den Handlungsfluss beeinträchtigt und die Glaubhaftigkeit des Avatar reduziert. Verzögerung ist auch vor allem dann ein Problem, wenn der Server des Avatar mehrere Aktivitäten von verschiedenen Nutzern koordinieren soll.

Für Adele wurde hier eine sehr leistungsfähige Lösung gefunden. Auf jeder Client-Maschine läuft eine separate Kopie von Adele. Die Ereignisse werden zwischen Adeles Maschinen aufgeteilt, wodurch kurze Antwortzeiten erreicht werden. Außerdem wird ein Lösungsplan problembezogen heruntergeladen. Durch die Fokussierung auf ein

²⁶ Als Phoneme bezeichnet man alle Laute einer gesprochenen Sprache, die die gleiche bedeutungsunterscheidene Funktion haben

Problem können alle relevanten Interaktionen statt finden, ohne dass Speicher und Prozessorleistung auf Client-Seite übermäßig beansprucht werden.

Zusammenfassend basieren animierte digitale Avatare auf einem intelligenten Tutoring-System. Hierfür benötigt werden:

- ein Expertenmodell, das das Wissen zur Problemlösung beinhaltet,
- ein Lernermodell mit allen Informationen über den Lerner in der aktuellen Situation,
- ein Verhaltensmodell, das ein Set mit allen Strukturen beinhaltet, die zur Steuerung des Lernprozesses eingesetzt werden und
- ein Kommunikationsmodell zur Steuerung der Interaktionen des Nutzers mit der Lernumgebung .

2.3.4 Vor- und Nachteile des Einsatzes virtueller Charaktere im Lernprozess

Digitale Agenten werden immer intelligenter und anspruchsvoller gestaltet. Trotzdem ist nach wie vor das Akzeptanzproblem nicht gelöst. Zum einen besteht eine gewisse Scheu und Zurückhaltung, da befürchtet wird, dass Lernen mit Avataren das Lernen im herkömmlichen Sinne verdrängen und Lehrkräfte ersetzen könnte. Diese Befürchtung ist jedoch weitgehend unbegründet, da Agenten in der Tutorenrolle nie dieselbe Autorität entgegengebracht wird, wie realen Lehrern. Das Vertrauen in einen Lehrkörper, das Grundvoraussetzung für eine erfolgreiche Kooperation ist, kann nur durch direkten sozialen Kontakt aufgebaut werden.

Umproblematisch ist die Akzeptanz von digitalen Agenten in den Rollen von Teammitgliedern oder als Repräsentation von realen Lehrkörpern und Mitlernern, da hier der Lerner keine Befehle von einer Maschine entgegennehmen muss.

Vorteile bringen Avatare vorallem im Zusammenhang mit Simulationen. Hier bieten sie den Nutzern die Möglichkeit Situationen realitätsnahe zu proben. Der Lerner kann seine Grenzen austesten, ohne Gefahr zu laufen realen Schaden anzurichten. Außerdem ist es

eine gute Möglichkeit um Gruppenarbeit zu koordinieren und zu optimieren. In diesem Zusammenhang kann es auch von Vorteil sein, wenn ein computergesteuerter Agent einen nicht anwesenden Lerner ersetzt, um die Gruppenarbeit trotzdem zu gewährleisten. Das gemeinsame Nutzen einer Umgebung schafft außerdem ein Zusammengehörigkeitsgefühl der Teammitglieder.

Als weiterer Vorteil kann in der Kommunikation gesehen werden. In der Arbeit mit einer Chatbot-basierten Anwendung lernt der Nutzer ganz nebenbei das gute und schnelle Formulierung, sowie die schnelle Eingabe über die Tastatur.

Als Repräsentation der eigenen Person in einer virtuellen Welt kann ein digitaler Agent die Selbstwahrnehmung des Nutzer stärken. Seine Ausdrucksweise und sein Verhalten werden in der virtuellen Welt umgesetzt und treffen möglicherweise auch auf direktes Feedback, was den Nutzer sensibler im Umgang mit seinen Mitmenschen macht.

Vor allem sollen digitale Agenten durch ihre Anwesenheit motivieren. Oftmals ist die bloße Anwesenheit einer Autorität - unter der Voraussetzung, dass der Avatar als eine solche anerkannt wird - Grund für verstärktes Engagement des Lerners im Lernprozess. Dieses Phänomen wurde durch zahlreiche amerikanische Studien bestätigt²⁷.

Nachteilig sind vordergründlich die sehr hohen Produktionskosten. Für leistungstarke animierte Agenten müssen komplexe Systeme der künstlichen Intelligenz implementiert werden, was enormes Know-How voraussetzt. In diesem Backend müsste je nach Größe und Umfang der Anwendung massive Mengen von Wissen implementiert werden, auf das der Avatar dann zur Laufzeit zurückgreifen kann. Wäre diese Wissensbasis nicht vorhanden, würde der gewünschte Lernerfolg verfehlt.

Der richtige Einsatz nonverbaler Signale bedeutet zusätzlichen Implementierungsaufwand. Außerdem ist die Generierung einer natürlichen Sprache mit Betonung und Satzmelodie immer noch in den Kinderschuhen.

27 Einen Überblick über die Untersuchungsergebnisse bietet (Bendel 2003b).

Besonders für junge Nutzer kann ein Avatar vom eigentlichen Inhalt, dem Lernen, ablenken, weil er zu sehr auf die Interaktion mit dem virtuellen Charakter konzentriert ist. Generell ist die Handhabung kritisch zu sehen. Zu Beginn kann es häufig zum wahllosen Herumklicken kommen, bis der Lerner weiß, welche Aktion die gewünschte Reaktion beim Avatar hervorruft. Auch die Kommunikation muss erst gelernt werden, da der Wortschatz auf Seiten des Avatar stark begrenzt ist.

Alles in Allem können virtuelle Agenten nur dann Lernerfolg bringen, wenn sie sorgsam und als Ergänzung zu realen Lernsituationen eingesetzt werden. Außerdem sollte die Entwicklung eine enge Zusammenarbeit von Informatikern, Pädagogen und Fachexperten bedeuten.

Abschließend soll bemerkt werden, dass es neben den animierten, hier vorgestellten Agenten auch nicht-animierte gibt, die in der Rolle eines Tutor oder Berater ebenso hilfreich sein können, um dem Lerner zu führen und diesen mit Fachwissen und Feedback zu versorgen.

2.4 Interaktivität

Oft wird bereits die Navigation innerhalb von Lernprogrammen als interaktiv bezeichnet. Eine Abgrenzung beider Begriffe ist in der Tat nur schwer zu treffen. Der Begriff Interaktion bezeichnet das Handeln mit Objekten, Gegenständen oder Inhalten. Die Navigation dient zur Steuerung des Ablaufes und der Auswahl der betrachteten Inhalte. Jedoch kann man durch die Bewegung eines Avatar den Prozess steuern, also Navigieren, wobei die eigentliche Handlung eine Interaktion mit dem Avatar verlangt.

Im folgenden sollen sechs Stufen der Interaktivität nach Schulmeister (2005) näher betrachtet werden um dessen Grad innerhalb von Lernprogrammen einzuordnen. Diese Betrachtung bezieht sich auf die zehnstufige „Guerra Scale“²⁸, die zwar etwas differen-

28 <http://www.learningcircuits.org/2004/mar2004/>

zierter ist, dadurch aber auch die Grenzen zwischen den einzelnen Stufen undeutlicher.

Betrachtet werden Multimedia- und Programmkomponenten eines Lernprogrammes. Zu den Multimedia-Komponenten zählen neben Bild und Ton auch Video und Animationen, Tabellen und Formeln oder JAVA-Applets. Programmkomponenten sind Programmsequenzen, also vordefinierte Teile des Programmes.

2.4.1 Stufe I: Objekte betrachten und rezipieren

Wie der Titel bereits vermuten lässt, werden Programmkomponenten und Inhalte bei dieser Stufe angeschaut und verinnerlicht. Neben dem Auf- und Wahrnehmen hat der Nutzer jedoch keine Möglichkeit den Ablauf zu beeinflussen oder Inhalte zu verändern. Übertragen auf die Praxis bezeichnet das den Start eines Programmes. Nach dem Start hat der Nutzer jedoch keinerlei Einfluss mehr; der Ablauf erfolgt automatisch. Wegen der fehlenden Handlungsvielfalt könnte man hier auch von einer Stufe 0 sprechen, da es keine direkte Interaktion gibt, sondern es sich um reine Navigation des Prozesses handelt.

Trotzdem ist diese Form nicht unbedeutend. Sachverhalte, Prozesse und Strukturen können auf diese Weise visualisiert werden. Außerdem sind solche Komponenten häufig Teil von Lernwelten, um dem Nutzer neben Text eine alternative Präsentation von Wissen zu bieten.

2.4.2 Stufe II: Multiple Darstellungen betrachten und rezipieren

Auch auf dieser Stufe ist der Inhalt fest vordefiniert. Jedoch kann der Nutzer, anders als auf der ersten Stufe, zwischen verschiedenen Darstellungsoptionen wählen. Das heißt, er kann einzelne Programmkomponenten durch das Einstellen vordefinierter Optionen oder Menüs verändern. Multimedia-Komponenten haben in Programmen dieses Interaktionsgrades meist nur illustrativen und informativen Charakter. Der Nutzer kann

durch die Wahl von Optionen zwischen verschiedenen Abbildungen und Varianten wählen, jedoch sind alle Varianten fest vordefiniert.

Darüber hinaus können die Programm-Sequenzen immer wieder wiederholt bzw. in beliebiger Reihenfolge angewählt werden. Der Lerner kann also den Programmablauf und damit den Lernprozess selbst steuern, hat jedoch keinen direkten Einfluss auf den Inhalt.

Der Vorteil liegt hier in den vielfältigen Abbildungsmöglichkeiten der Realität. Im Vordergrund steht immernoch die Veranschaulichung von Prozessen und Strukturen, wobei der Lerner in dieser Stufe bereits einigen Einfluss hat, indem er unter verschiedenen Alternativen wählen kann. Dadurch wird das kritische Denken geschult, der Lerner vergleicht, entdeckt Gegensätze und Kontraste, sowie Widersprüche und auch Übereinstimmungen. Die Betrachtung eines Gegenstandes erfolgt aus mehreren Sichten.

2.4.3 Stufe III: Die Repräsentationsform variieren

Auf dieser Stufe hat der Lerner direkten Einfluss auf den Inhalt, d.h. auf die Multimedia-Komponenten. Bilder können skaliert oder gedreht werden, 3D-Objekte können „angefasst“ und so von allen Seiten betrachtet werden und ein Video enthält interaktive Elemente zur Steuerung des Ablaufes. Bei Programmen ist es vorstellbar, dass es mehrere Varianten gibt, z.B. mit unterschiedlichen Schwierigkeitsgraden oder durch die Wahl verschiedener Szenarien. Der Inhalt und auch der Ablauf der Programmvarianten ist jedoch jedes für sich fest vorprogrammiert und somit nicht manipulierbar.

Die direkte Interaktion des Lerners mit den Inhalten motiviert. Ist es jedoch nicht mit einer konkreten Fragestellung verbunden, kann es sehr schnell zu einer Spielerei werden, die keinerlei Nutzen für den Lerner hat, außer ihn abzulenken. Hat die Lerneinheit jedoch zum Ziel die Eigenschaften eines Gegenstandes zu lernen, ist es sehr

wertvoll sein Verhalten durch umseitige Betrachtung testen zu können.

2.4.4 Stufe IV: Inhalt der Komponenten beeinflussen

Die Inhalte sind nicht fest vordefiniert, sondern werden erst aus den Nutzereingaben generiert. Der Nutzer kann in einem vordefinierten Rahmen Parametereinstellungen manipulieren und so die Darstellungsform bestimmen. Auch Programmsequenzen können durch Manipulation der Einstellungen, z.B. dem Aufruf verschiedener Datensätze, in ihrem Inhalt variieren. So kann der Lernende beispielsweise in einem Geometrie-Programm aus einer vordefinierten Menge eine Funktion wählen, zu welcher dann der Graph vom Programm generiert wird. Durch Veränderung der Parameter, kann man nun den Graphen verändern, z.B. ihn Strecken oder Stauchen .

Vorraussetzung sind an dieser Stelle entsprechende Vorkenntnisse des Lerner, um sinnvolle Einstellungen zu wählen. Der Nutzer eines Geometrie-Programmes sollte wissen, wie man einen Graphen beschreibt um zu einem Ergebnis zu gelangen. Sind entsprechende Vorkenntnisse vorhanden eignet sich solche Software, um eigene Ideen und Hypothesen zu überprüfen. Außerdem können vorallem abstrakte Sachverhalte, wie beispielsweise mathematische Formeln, veranschaulicht werden.

2.4.5 Stufe V: Objekt bzw. Inhalt der Repräsentation konstruieren und Prozesse generieren

Programme dieser Interaktivitätsstufe sind Werkzeuge, mithilfe dessen der Lerner seine Ideen und Gedanken visualisieren kann. Ein Geometrie-Pogramm würde dem Lerner an dieser Stelle erlauben jede beliebige Funktion anzugeben, aus dieser das Programm dann den Graphen konstruiert.

Auch hier sind Vorkenntnisse über den jeweiligen Sachverhalt notwendig, da es sonst schnell zur Frustration führen könnte. Wenn man keine Kenntnisse über die

Beschreibung eines Graphen hat und deshalb sinnlose Eingaben macht, kann das Programm diese auch nicht auswerten und somit kein Ergebnis liefern.

2.4.6 Stufe VI: Konstruktive und manipulierende Handlungen mit situationsabhängigen Rückmeldungen

Anders als in Stufe V gibt das System in Stufe VI zusätzlich sinnvolle Rückmeldung zu den Nutzereingaben. Am Beispiel des Geometrie-Programmes bedeutet das, dass nun beliebige Nutzereingaben vom System ausgewertet werden können. Sollten die Eingaben nicht den syntaktischen Aufbau mathematischer Formeln folgen, kann das System auch darauf reagieren, indem er den Lerner darauf aufmerksam macht und ihm Hilfestellung zur richtigen Eingabe gibt.

“Interaktivität auf dieser Stufe bedeutet zusammenfassend, dass dem „Partner“ Computer oder Programm bedeutungstragende Objekte bzw. Aktionen zugeschickt werden, die das Programm versteht und auf das das Programm mit entsprechend bedeutungsvollen Handlungen antworten kann.“ (Schulmeister 2005)

Hiermit ist der höchste Grad an Interaktivität erreicht. Lernprogramme mit intelligenten Avataren können ebenfalls dieser Interaktionsstufe zugeordnet werden. Die Software ist in der Lage, auf jede Eingabe eine sinnvolle Antwort zu haben. Solche Programme gibt in der Realität nicht, da es unmöglich ist, alle Eventualitäten im Vorfeld abzufangen. Deshalb wird meist eine Standard-Reaktion implementiert, die ausgeführt wird, wenn die Software kein geeignetes Verhalten auf die Eingaben findet.

2.5 Persistenz

Die Persistenz eines Systems beschreibt dessen Eigenschaft, Daten dauerhaft speichern zu können. Das ist vorallem für Lernanwendungen wichtig, die in mehr als einer Sitzung bearbeitet werden. Ist die Datenspeicherung möglich, kann der Lerner bei

erneutem Start des Programmes an seinen letzten Lernfortschritt anschließen, anstatt wieder von vorne zu beginnen.

Die Datenspeicherung und Verwaltung wird meist von einer Datenbank übernommen, mit der das Programm kommuniziert. Hier können alle Daten abgelegt und zur Laufzeit wieder abgerufen werden. Die gängigsten Autorensysteme, wie z.B. Flash bieten die Möglichkeit Verbindung zu einer Datenbank herzustellen und dessen Einträge auszulesen und gegebenenfalls auch zu überschreiben.

Wenn mehrere Nutzer dieselbe Anwendung nutzen, z.B. wenn die Anwendung online verfügbar ist oder als offline-Variante von mehreren Nutzern auf einem Rechner genutzt wird, müssen die Daten dem aktuellen Lerner zugeordnet werden können. Dazu muss sich der Nutzer nach dem Programmstart identifizieren. Mithilfe der Login-Daten wird dann der entsprechende Datensatz aus der Datenbank gesucht und dem System übergeben. Dieses kann anhand der Daten den letzten Status der Welt rekonstruieren.

Das setzt voraus, dass die Daten am Ende einer Anwendung gespeichert werden. Dies sollte möglichst automatisch funktionieren, falls der Lerner das manuelle Auslösen des Speicherbefehls vergisst. Weiterhin wäre es wünschenswert, wenn das System zur Laufzeit automatisch die Datensätze aktualisieren würde, um auch bei einem unerwünschten Beenden des Programms, z.B. durch einen Stromausfall, das Bestehen des Lernfortschrittes zu gewährleisten.

2.6 Anwendung der Charakteristika zur Typisierung von Lernprogrammen

Nachdem in den vorigen Unterkapiteln die Eigenschaften näher erläutert wurden, die eine virtuelle Lernwelt mit sich bringt, wird nun mithilfe dessen in diesem Abschnitt eine Gruppierung der unzähligen Programmtypen vorgenommen. Dazu wurden sechs Typen gebildet. Betrachtet werden Hypermedia-Systeme, (intelligente) Tutoring-Systeme und Übungspogramme, Simulationen, Lernkonferenzsysteme, Edutainment

Programme und Soziale Welten.

2.6.1 Hypermedia-Systeme

Wie sich aus dem Namen erahnen lässt, handelt es sich hierbei um reine Informationssysteme im Internet. Das Wort Hypermedia besteht aus dem Begriff Hypertext mit deutlicher Ausrichtung auf Multimedialität (Wikipedia 2009c). Hypertext deutet auf eine netzartig aufgebaute Informationsstruktur hin, die aus Knoten besteht, welche miteinander verknüpft sind. Dabei enthalten die Knoten multimediale Inhalte, d.h. Text, Bild, Video und Ton bilden die Informationsobjekte. Die Suche innerhalb solcher Systeme verläuft nicht-linear. Es gibt keinen Start und Endpunkt, sondern unendlich viele verschiedene Wege um an eine Information zu gelangen. Hypermedia-Systeme werden häufig im Bereich Open Content eingesetzt. Das heißt, jeder kann sich am Ausbau der Informationsstruktur durch das Hinzufügen von Artikeln beteiligen.

Eines der wohl bekanntesten Hypermedia-Systeme ist Wikipedia²⁹. Man kann durch Eingabe von Stichwörtern nach Informationen suchen. Passend zu den Eingaben erhält man Artikel, in denen diese Stichwörter von Bedeutung sind. Hat man sich nun für einen Artikel entschieden, ist dieser wiederum mit zahlreichen Verlinkungen gespickt, die zu anderen Einträgen weiterleiten. Die Artikel sind mit Bildern und stellenweise auch Videos gespickt.

Diese Systeme sind besonders zur Aneignung von spezifischen Wissen geeignet. Ganz nebenbei erhält man Zusatzinformationen, die gegebenenfalls zur Vertiefung des Themas nützlich sind.

Das größte Problem von Hypermedia-Systemen mit Open Content wie Wikipedia ist die Glaubwürdigkeit der Artikel. Da es sich dabei um ein Netzwerk handelt, dass jeder Internetnutzer nicht nur nutzen, sondern auch erweitern kann, kommt es leicht zu

29 <http://de.wikipedia.org/wiki/Wikipedia:Hauptseite>

fehlerhaften Inhalten. Jedoch ist ein großes Team von überwiegend freiwilligen Autoren rund um die Uhr bemüht, die Artikel auf ihre inhaltliche Richtigkeit zu prüfen.

Hypermedia-Systeme sind in der Regel 2D-Anwendungen. Die Kommunikation mit dem System erfolgt wie eine Chatbot über Texteingaben und -ausgaben. Jedoch begrenzt sich der Dialog auf die Suche nach spezifischen Wissen.

In dieser Kategorie wären Avatare vorwiegend als Navigationshilfen denkbar. Sie könnten die Schnittstelle zwischen Nutzer und System bilden. In diesem Fall würde der Nutzer dem digitalen Agenten seine Anfrage mitteilen, welcher dann die Suche nach passenden Einträgen anleitet.

Der Interaktionsgrad in Systemen wie Wikipedia ist relativ hoch, da der Nutzer die Inhalte nicht nur betrachten, sondern auch verändern kann.

Hypermedia-Systeme können nur begrenzt zu den virtuellen Lernwelten gezählt werden, sondern eher zu den Informationssystemen. Hier wird nur Wissen gesammelt und für eine große Menge von Nutzern zugänglich gemacht. Jedoch gibt es keine Möglichkeit Wissen durch Übungen o.ä. zu testen.

Da die Informationen netzwerkartig angeordnet sind und die Navigation durch das System absolut beliebig erfolgt, gibt es keinen Lernprozess der gespeichert werden könnte.

Hypermedia-Systeme sind häufig Teil komplexer virtueller Lernumgebungen. So werden sie zum Beispiel in Form von Bibliotheken zur gezielten Suche nach Informationen eingesetzt.

2.6.2 (intelligente) Tutoringsysteme und Übungsprogramme

Diese beiden Typen werden hier einer Gruppe zugeordnet, da sie die typischen und bekannten eLearning-Programme beschreiben. Zu dieser Kategorie gehören virtuelle Lernwelten genauso wie „einfache“ CBT- und WBT-Anwendungen.

Tutoringsysteme und Übungsprogramme dienen der Wissensvermittlung und -abfrage.

In Modulen wird dem Lerner linear handlungsbezogenes Wissen dargeboten. Hierzu können Texte und Bilder, genauso wie kleine Animationen, Ton und Videos in die Anwendung eingebunden sein. Auf die Module kann eine Wissensabfrage folgen. Das System stellt dem Lerner stoffrelevante Fragen, der Lerner beantwortet diese und bekommt ein vordefiniertes Feedback.

Eine Weiterentwicklung traditioneller Tutoringsysteme sind intelligente tutorielle Systeme. Auf Grundlage der Nutzereingaben entscheidet das System zur Laufzeit über den Lernweg, den der Nutzer gehen soll. Auf diese Weise findet eine Individualisierung der Lernsoftware auf Basis der Kenntnisse des Lerners statt. Das Lernermodell wird vom System gespeichert und mit fortlaufender Entwicklung angepasst. In dieser Kategorie werden immer häufiger Avatare eingesetzt, um den Lerner zu unterstützen. Sie können die Rolle eines aktiven Tutors einnehmen, der den Lerner anleitet, aber ihm auch einfach bei seinem Lernprozess zur Seite stehen und bei Fragen aktiv werden.

Handelt es sich um intelligente Tutoringsysteme, so erreicht der Interaktionsgrad der Anwendung die höchste Stufe. Das System gibt, beispielsweise über einen animierten Avatar, auf Nutzereingaben sinnvolle Rückkopplung. Auch bei nicht-intelligenten Systemen ist der Einsatz von digitalen Agenten durchaus sinnvoll, um den Lerner durch den Lernprozess zu begleiten und ihn gegebenenfalls mit Zusatzinformationen zu versorgen oder vordefiniertes Feedback zu geben.

Bei dieser Gruppe von Lernanwendungen gibt es in der Regel ein Lernmanagementsystem (LMS) im Hintergrund, von dem die Nutzerdaten und Daten zu dessen Lernfortschritt verwaltet werden. Außerdem stellt das LMS die Inhalte und dessen Struktur (z.B. in Kursen) bereit. Über den Webbrowser können Lerner die Inhalte nutzen. Außerdem gibt es in solchen Programmen häufig Möglichkeiten, wie Foren, E-Mail oder Chat, um sich mit Trainern oder anderen Lernern auszutauschen. Die Persistenz spielt hier eine wichtige Rolle um den Lerner einen reibungslosen Lernprozess bieten zu können.

Übungsprogramme und Tutoringsysteme sind bis jetzt zum großen Teil in 2D realisiert.

Das ist begründet mit dem bis vor kurzen herrschenden Darstellungsschwierigkeiten von 3D-Objekten im Web (2.2.2 3D). Auch heute muss eine gute Bandbreite anliegen, um einen schnellen Aufbau der Lerninhalte zu gewährleisten. Sollte dies nicht der Fall sein, führen die langen Wartezeiten beim Laden der Inhalte zu Frustration beim Nutzer.

Programme dieser Gruppe finden heutzutage in fast allen Bereichen Anwendung. Besonders in der beruflichen Aus- und Weiterbildung werden sie eingesetzt und bieten die Möglichkeit der zeit- und ortsunabhängigen Wissensaneignung, sowie der Schulung großer Lernermengen. Beispiele sind Produktschulungen, Programme zum Sprachenlernen oder Softwareschulungen. Besonders beliebt sind Lernprogramme dann, wenn eine Schulung durch eine große Teilnehmerzahl oder eine weite Streuung der Teilnehmer sehr kostenintensiv wäre.

2.6.3 Simulationen

In Simulationen werden reale Umgebungen und Situationen nachgestellt, meist in Form von 3D-Welten. Dort können sich die Nutzer bewegen und aus verschiedenen Perspektiven umschauen. Der Lernende kann seine eigenen Fähigkeiten austesten und erfährt sofort die Konsequenz seines Handelns. Besonders geeignet sind Simulationen zum Teamtraining. Der Nutzer kann in verschiedene Rollen schlüpfen und somit aus verschiedenen Perspektiven ein Problem betrachten. Die eigene Person, sowie andere Übungsteilnehmer werden in Simulationen durch virtuelle Agenten repräsentiert. Dabei müssen diese kein intelligentes Verhalten an den Tag legen, sondern sich entsprechend der Nutzereingaben bewegen. Nicht intelligent bedeutet in diesem Zusammenhang, dass sie den Nutzer nicht mit Wissen versorgen müssen.

Simulationen dienen vordergründig dem Trainieren von Extremsituationen und zum Experimentieren. Es können lebensbedrohliche Szenarien durchgespielt werden, aber auch solche, deren Nachstellung sehr aufwendig und kostenintensiv wäre. Die Umgebung ist sicher für alle und vorallem reproduzierbar, d.h. es entsteht kein dauerhafter Schaden.

Die Trainer haben sehr hohe Kontrolle über den Lernprozess, da sie durch Einstellungen der Umgebung genaue Kontrolle über die Situation haben. Diese Einstellungen können je nach Grad der Interaktion vor dem Start definiert (Stufe III) oder während des Trainings angepasst werden (Stufe IV).

Simulationen bringen auch ihre Nachteile mit sich. Keine reale Situation lässt sich in ihrem vollen Umfang digitalisieren. Außerdem fehlt ein wichtiger Faktor, der in realen Situationen hinzukommt - der Stressfaktor des Lernalters bei der Bewältigung schwieriger Situationen. Aufgrund dessen können Lernergebnisse verfälscht werden. Es kann den Nutzer in realen Situationen auch unvorsichtig werden lassen, weil er die Lage unterschätzt.

Besonders häufigen Einsatz finden Simulationen in der medizinischen Ausbildung und in Berufen, in denen das reale Training sehr gefährlich wäre. Zwei Beispiele, Adele und Steve, wurden in dieser Arbeit bereits vorgestellt. Auf der nachstehenden **Abbildung 2.8** sind zwei weitere Beispiele aufgezeigt.



Abbildung 2.8: links: RescueSim, rechts: INMEDIA

RescueSim³⁰ ist ein Produkt der Firma Vstep aus Rotterdam (NL). Es handelt sich dabei um eine interaktive 3D Trainings- / Simulationsumgebung zum Nachstellen und Proben von Notfalleinsätzen. Auf Abbildung 2.9 links ist zum Beispiel ein Trainingsszenario für Feuerwehrmänner zu sehen. INMEDIA³¹ ist ein Fallsimulator für die Aus- und

30 <http://www.rescuesim.com/>

31 <http://www.inmedea-simulator.net/>

Weiterbildung in medizinischen Berufen. Mithilfe der Software können typische Patienten und Krankheitsverläufe untersucht, diagnostiziert und behandelt werden. INMEDIA ist ein Produkt der INMEDIA GmbH in Reutlingen (D).

2.6.4 Lernkonferenzsysteme

Lernkonferenzsysteme, auch bezeichnet als computergestütztes kooperatives Lernen (engl. Computer Supported Cooperative Learning), unterstützen die Kommunikation zwischen Lehrer und Lernenden. Die Vorteile virtueller Welten werden genutzt, um eine optimale Zusammenarbeit und beste Unterstützung der Lernenden zu gewährleisten. Genutzt werden Foren und Chats bzw. die Möglichkeit des direkten Austausches in einer virtuellen Welt zwischen Avataren, die Lerner und Lehrer repräsentieren.

Zu den Lernkonferenzsystemen zählen auch Webinare. Das Wort beschreibt ein Seminar, das über das Web abgehalten wird. (Wikipedia 2009d)

Dabei gibt einen Moderator und die Teilnehmer, die über ein Programm den Bildschirm des Moderators sehen können und seinen Anweisungen folgen. Die Teilnehmer haben in der Regel die Möglichkeit über Chat oder VoiceOverIP Fragen an den Moderator zu stellen.

Eine weitere Variante von Lernkonferenzsystemen sind gemischte Realitäten. Hier erfolgt ein Austausch zwischen realer und virtueller Umgebung, um geografisch weit gestreute Lernende in „einem“ Raum zu bringen. Ein Beispiel ist das MiRTLE Projekt³². Dieses Projekt nutzt die virtuelle Welt-Software Project Wonderland von Sun Microsoft, die bereits unter *2.1.2 Sharable Content – Inhalte gemeinsam nutzen* vorgestellt wurde, um Studenten aus der International Academy der Essex University (UK) mit Studenten der Shanghai Jiao Tong University in China zusammen zu bringen.

Es gibt einen realen Klassenraum, der sich beispielsweise in China befindet. Dieser

32 <https://slx.sun.com/1179271385>

Raum ist mit einem großen Bildschirm ausgestattet (Abb. 2.9 linker Bildbereich), über die die abwesenden Lerner, also die Studenten in Essex, gesehen. Außerdem sind Kamera und Microphone installiert, die die anwesenden Lerner in China aufzeichnet.

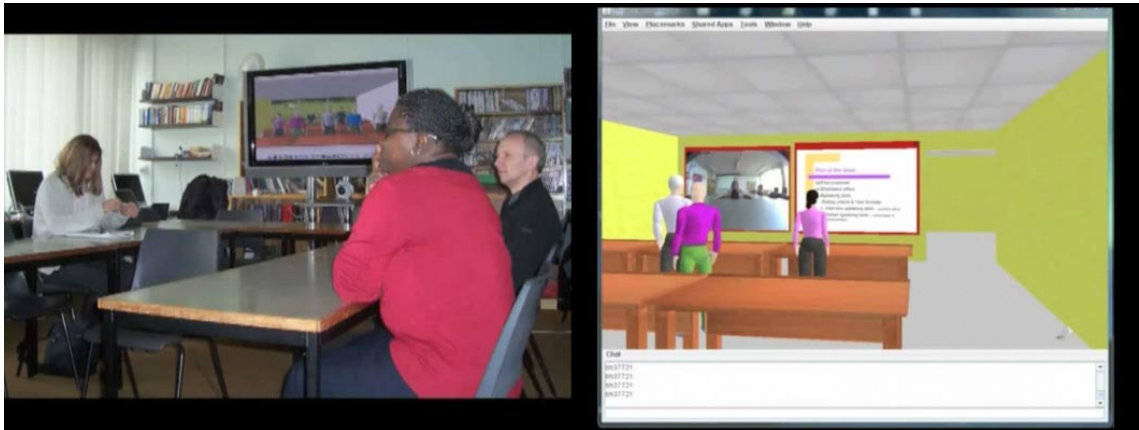


Abbildung 2.9: MiRTLE Projekt (Quelle: <https://slx.sun.com/1179271385>)

Die nicht anwesenden Lerner der Essex University befinden sich als Avatare in einer virtuellen Welt. Sie haben in der virtuellen Welt zwei Bildschirme, auf denen sie die Lerner im Klassenraum und die präsentierten Lehrmaterialien sehen können, wie auf Abbildung 2.10 im rechten Bildbereich zu sehen. Die abwesenden Studenten können mit den anderen über Chat oder Audio kommunizieren.

Lernkonferenzsysteme sind nur bedingt virtuelle Welten. Für Webinare wird eine spezielle Software genutzt, die Desktop Sharing³³ und die Kommunikation über Chat und VoiceOverIP erlaubt. Andererseits könnte auch eine virtuelle Welt Treffpunkt für die Teilnehmer des Seminars sein. Das Projekt MiRTLE nutzt die Möglichkeiten des geteilten Raumes um abwesende Studenten miteinzubeziehen und die verfügbaren Präsentations- und Kommunikationsmöglichkeiten einer virtuellen Welt. Die Interaktion der Lerner mit den Lerninhalten ist sehr beschränkt (Stufe I), da sie diese eigentlich nur betrachten können.

33 Beim Desktop Sharing kann ein Teilnehmer seinen Bildschirm für alle sichtbar machen, sodass diese sein Handeln verfolgen können. Bei Trainings ist das meist der Moderator.

2.6.5 Edutainment Programme

Das Wort Edutainment ist relativ neu, wird jedoch immer häufiger benutzt. Es entstand aus der Zusammenführung der Begriffe Education, englisch für Bildung, und Entertainment, was mit Unterhaltung übersetzt werden kann. Edutainment beschreibt folglich das spielerische Lernen. Synonyme sind Lernspiel oder Neudeutsch Game Based Learning.

Nach Schanda (1995) sind „*Lernspiele am Computer [...] Programme, die auf dauerhaften Erwerb bzw. die Anwendung von Fähigkeiten / Fertigkeiten abzielen*“. Auch für das Institut für angewandte Kindermedienforschung in Stuttgart (IFAK)³⁴ geht es bei Edutainment Games vordergründig um die Wissensvermittlung. Jedoch nicht unbeachtet bleiben soll der Spassfaktor beim Lernen mittels Edutainment Programmen, der laut IFAK stark variieren kann.

Vermittelt werden soll meist Faktenwissen, dass in spannenden Handlungen verpackt und im Spielprozess nebenbei erlangt wird. Die Kombination von Übungsphasen und spielerischen Elementen bieten besonders für junge Lerner ideale Voraussetzungen zum erfolgreichen Wissenserwerb. Jedoch führt der Spass am Lernen auch bei Erwachsenen zu mehr Motivation und damit einer nachhaltigen Wissensaneignung.

Für den einzelnen Lerner ist der Lernstoff je nach Interesse zugänglich. Die Abarbeitung erfolgt nicht modular und folgt somit keinem Lehrplan. Durch den Einfluss des Lerners auf die Spielhandlung, wird der Lernprozess unbewußt von ihm selbst gestaltet und verändert.

Eine der zur Zeit bekanntesten Edutainment-Projekte für Kinder ist die Lernplattform Scoyo³⁵ (Abb. 2.10). Diese wurde für Schüler der ersten bis siebten Klasse entwickelt und stofflich an die Lehrpläne der einzelnen Bundesländer Deutschlands angepasst. Der Nutzer kann sich ein Fach wählen und lernt den Stoff, verpackt in lustigen Geschichten

34 <http://hdm-stuttgart.de/ifak/>

35 <http://www.scoyo.de/>

mit den verschiedensten Charakteren in spannenden Welten. Zwischendurch wird er immer wieder aufgefordert sein Wissen anzuwenden, indem er Aufgaben, die aus der Geschichte entstehen, lösen soll und so seinen virtuellen „Freunden“ (gemeint sind hier die virtuellen Charaktere der Geschichten) hilft. Der Lernfortschritt wird gesichert, sodass der Lerner beim nächsten Einloggen auf seinen bisherigen Kenntnissen aufbauen kann. Natürlich kann alles beliebig oft wiederholt werden. Daraus lässt sich schließen, dass ein Lernmanagementsystem die Basis bildet, das die Verwaltung der Lernerdaten übernimmt und danach die Umgebung an den Lernfortschritt des Nutzers anpasst. Die Nuzeroberfläche ist 2D, wobei in den Navigationselementen der Lernplattform eine Plastizität angedeutet wurde. Die direkte Lernumgebung, d.h. die kleinen virtuellen Welten, in denen sich die Geschichten abspielen, sind im Comic-Stil gestaltet.

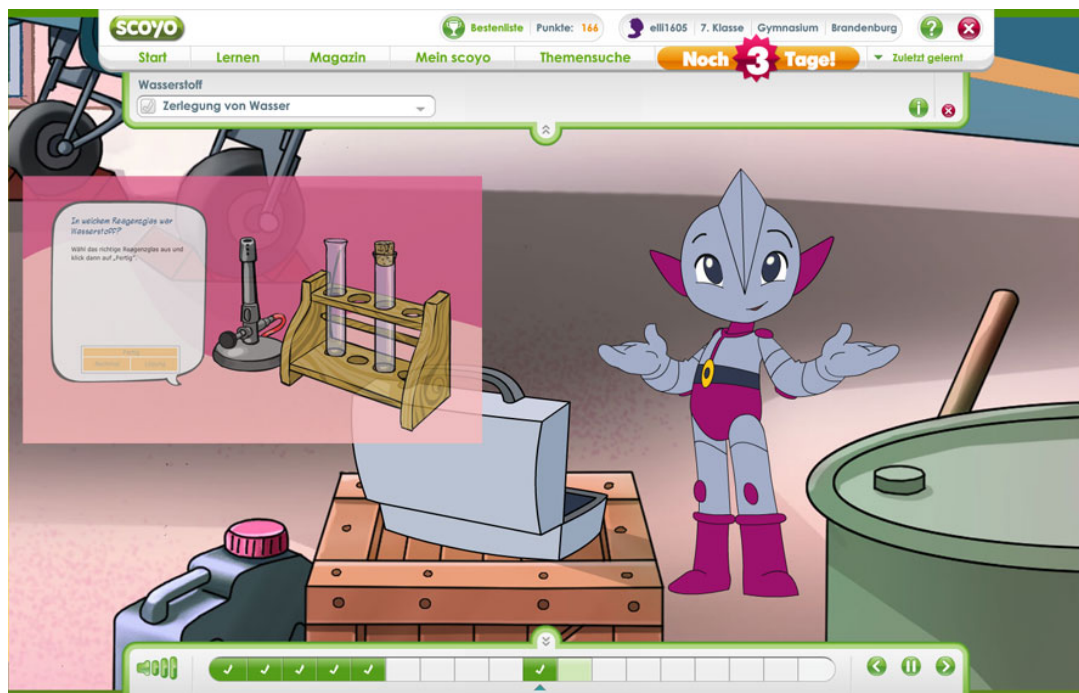


Abbildung 2.10: Scoyo (Quelle: <http://www.scoyo.de/>)

Eine Gruppe der Edutainment Programme sind die Serious Games. Oftmals wird Serious Games als Synonym für Lernspiel verwendet, da es sich auch hier um Softwareprodukte zum spielerischen Wissenserwerb handelt. Dennoch ist das nicht ganz berechtigt. Serious Games sollen vor allem Anderen Spass und Unterhaltung bieten. Kombiniert ist das Ganze mit ernstern Absichten zur Wissensvermittlung. Realisiert wird das, indem die Umgebung bzw. Spielaufgaben auf Probleme oder Umstände des Alltags

oder die Situation in anderen Ländern aufmerksam machen und ein Bewußtsein dafür schaffen .

Nach Chen & Michael (2006) können Serious Games nochmals unterteilt werden. Die wichtigsten Gruppen sind Military Games, Government Games und Healthcare Games. Serious Games aus dem Militärbereich gibt es schon seit 1980. Diese werden zur Ausbildung der Rekruten eingesetzt, da reale Trainings zu gefährlich und kostenintensiv wären. Weiterhin gibt es die Government Games, die zum Krisenmanagement in sozialen und wirtschaftlichen Bereichen, oder zur Verkehrs- und Städteplanung eingesetzt werden. Auch politische Belange werden in Government Games behandelt. Die letzte große Gruppe sind die Healthcare Games. Diese sind vor allem für Kranke und die Menschen in ihrer Umgebung. Dadurch wird unbewusst die Fähigkeit des besseren und bewussteren Umgangs mit der eigenen Krankheit oder der eines Bekannten vermittelt. (vgl. Chen & Michael 2006)

Der wesentliche Unterschied zwischen Edutainment Games und Serious Games ist die Dosierung der Übungs- und Spielphasen. Edutainment Programme dienen der Bildung, wobei der Stoff in spielerischer Form vermittelt wird und so vor allem für junge Lerner besser erfassbar ist. Es gibt ein festes Lernziel. Serious Games sind Spiele, in denen hintergründiges Wissen nebenbei vermittelt werden soll. Die Handlungen sind also nicht fiktiv, sondern weisen auf regionale oder nationale Gegebenheiten hin. Ein Lernziel gibt es nicht direkt. Es geht vielmehr darum geeignete Strategien zu finden, um Situationen zu bewältigen.

Der Interaktionsgrad ist in beiden Fällen recht hoch, da sich der Verlauf den Nutzeraktivitäten anpasst. Avatare können in beiden Fällen eine Rolle spielen, z.B. als eigene oder andere Spielfiguren in Serious Games oder als Berater und Guide in Edutainment-Programmen. Sie sind aber nicht zwingend erforderlich. Die Nutzeroberfläche kann je nach Zielgruppe und Verwendung individuell gestaltet werden. Wie schon im oberen Teil erwähnt, ist es Teil der Software, dass Lerner an den letzten Stand ansetzen können. Deshalb ist ein persistentes System zur Verwaltung der Nutzerdaten sehr wichtig.

2.6.6 Soziale Welten

Unter soziale Welten, oder Second Life, werden hier die Vergnügungswelten für Jung und Alt zusammen gefasst. Die Zahl der virtuellen Communities steigt täglich, genau wie die Zahl der registrierten Nutzer. Menschen nutzen das Internet, um sich zu treffen, sich auszutauschen und Neues kennen zu lernen. Solche Welten gibt es für jedes Alter, wobei diese sich vor allem in ihrem Erscheinungsbild unterscheiden (vgl. 2.2 *Grafische Nutzeroberfläche*). Diese Kategorie soll hier nicht näher erläutert werden, sondern nur der Vollständigkeit halber erwähnt, da es in dieser Arbeit vordergründig um virtuelle Welten mit dem Fokus Bildung geht.

2.7 Zusammenfassung / Mehrwert virtueller Lernwelten

Im Laufe des Kapitels wurden alle wesentlichen Merkmale erläutert, die virtuelle Welten auszeichnen. Diese sind in unterschiedlichem Ausmaß in den einzelnen Programmtypen wieder zu finden. Alle Typen können für sich alleine eine virtuelle Welt bilden oder aber zusammen mit anderen Komponenten Teil einer virtuellen Lernwelt sein.

E-Learning, d.h. das Lernen mit elektronischen Inhalten, erfreut sich wachsender Beliebtheit. Die Gründe dafür liegen auf der Hand. Man kann lernen wann und wo man will. Durch die rasche Ausbreitung des Internets wird es immer leichter Materialien für große Mengen von Nutzern zugänglich zu machen. Mit virtuellen Lernwelten wird den Lerninhalten nun eine hübsche Verpackung gegeben. Damit ist nicht eine besonders aufwendige Gestaltung gemeint, sondern vielmehr die Schaffung einer Umgebung, in der sich der Lerner bewegen kann. Die Welt bietet zum einen Raum für die Themen, die behandelt werden sollen, und zum anderen erzählt sie eine Geschichte, die den Lerner fesselt. Die Idee ist es eine Atmosphäre zu schaffen, in der das Lernen zum Erlebnis wird.

Die Umgebung kann dabei ganz unterschiedlich ausschauen, oft ist es jedoch die

Abbildung realer Gebäude und Umgebungen. Diese Abbildungen können wirklichkeitsgetreu sein, d.h. virtuelle Modelle realer Gebäude, aber auch metaphorisch. Ist die Lernwelt zum Beispiel einem Campus nachempfunden, können elementare Bereiche in der virtuellen Umgebung nachgebildet werden. Es gibt Vorlesungssäle, in denen die Kurse stattfinden, eine Bibliothek, hinter der eine Literaturlatenbank implementiert ist oder die auf ein Hypermedia-System verlinkt, eine Mensa, in der man sich mit anderen Lernern treffen und sich austauschen kann, z.B. via Chat, und eine Information, die dem Lerner bei der Orientierung innerhalb der Lernumgebung hilft.

Durch die räumliche Struktur gelangt der Lerner schnell und intuitiv zu seinem Ziel. Dadurch erfolgt eine Identifizierung mit der Welt, was das Lernerlebnis nachhaltig werden lässt. Jeder Lerner hat die Möglichkeit seinen eigenen Weg durch die virtuelle Welt zu gehen und bestimmt somit selbst Tempo und Reihenfolge bei der Bearbeitung des Stoffes. Durch die Darstellung räumlicher Strukturen kann auch der Stoff besser gegliedert und in kleinen Einheiten präsentiert werden. Auch das trägt zur Nachhaltigkeit des Lernerlebnisses bei.

Durch die vielfältigen Darstellungsmöglichkeiten in und Interaktionen mit der Welt, werden alle Lerntypen (visuelle, auditive und taktile) angesprochen. Durch Kombination visueller, auditiver und taktile Reize kann eine Gedächtnisleistung von bis zu 90 % erreicht werden - das Lernen mit allen Sinnen führt zu maximalen Erfolgen.

Wie schon in Kapitel 2.3 *Direktheit / Unmittelbarkeit* beschrieben, tragen auch Avatare zu einem nachhaltigen Lernerfolg in einer virtuellen Lernwelt bei. Schon durch die bloße Anwesenheit von virtuellen Charakteren als Repräsentation von anderen Lernern oder Tutoren steigt die Disziplin beim Lernenden. Außerdem entsteht eine Gruppendynamik, die sich positiv auf die Motivation des Einzelnen auswirkt. Gartner (2007) sagt voraus, dass bereits in zwei Jahren, Ende 2011, 80 % aller Internetnutzer Avatare haben werden und in mehreren (sozialen) Welten registriert sind. Dennoch sind echte soziale Kontakte im Bildungskontext unentbehrlich.

Trotz all der neuen Möglichkeiten für die Bereicherung des Unterrichts ist der enorme

Entwicklungsaufwand für virtuelle Lernwelten nicht zu vernachlässigen. Es bedarf neben speziellem technischen Know-How auch hohen Ressourcenaufwand. Auch wenn teure Seminare eingespart werden, lohnt sich die Entwicklung aus Kostensicht erst, wenn die Lehrinhalte für längere Zeit aktuell sind und die Plattform von vielen Lernern genutzt wird. Zusätzlich scheint es sinnvoll geschultes Betreuungspersonal einzusetzen um Anwendern, die wenig Erfahrung im Umgang mit multimedialen Programmen haben, beim Einstieg zu unterstützen und Frustrationserlebnisse zu vermeiden.

3. Lernen via iPhone

3.1 Mobile Learning

Der Begriff mobiles Lernen, auch als M-Learning, wireless oder pervasive Learning bezeichnet, leitet sich vom Begriff E-Learning ab und beschreibt das Lernen mittels portabler, drahtlos vernetzter Endgeräte. (Ilmenau 2009)

Der drahtlose Internetzugang und ständig besser werdende Akku-Laufzeiten machen das Notebook zu einem M-Learning-fähigen Endgerät. Außerdem bringt es ähnliche Prozessorleistungen und Speicherumfang mit, wie es von normalen PCs und Laptops bekannt ist. Die Schnittstellen sind dieselben, sodass die Bedienung nicht extra erlernt werden muss. Es handelt sich letztendlich um eine möglichst kleine Ausführung eines PCs mit ähnlich großem Leistungsumfang. Aus diesen Gründen müssten E-Learning-Anwendungen nicht extra angepasst werden um sie mobil nutzbar zu machen. Dennoch sind Notebooks wegen ihrer Größe und einer begrenzten Akku-Leistung relativ unhandlich für das Lernen unterwegs. Außerdem sind die Geräte in ihrer Anschaffung noch relativ teuer.

Auch Handys scheinen sich wunderbar zum mobilen Lernen zu eignen. Sie sind in der heutigen Zeit ständiger Begleiter und deshalb immer verfügbar. Außerdem werden sie bereits von Kindern und Jugendlichen genutzt. Dennoch eignen sie sich nur für sehr reduzierte M-Learning-Anwendungen. Grund dafür ist zum einen das kleine Display, dass das Lesen bzw. konzentrierte Betrachten auf längere Zeit sehr anstrengend macht. Weiterhin können sie mit einer weit geringeren Rechen- und Speicherleistung aufwarten im Vergleich zu Notebooks und PCs. Beinahe täglich wächst der Markt der Handy-anbieter. Die Vielzahl der verfügbaren Modelle macht die Erstellung allgemein nutzbarer Anwendungen beinahe unmöglich. Außerdem ist die Programmoberfläche vordefiniert und nur begrenzt durch Installation eigener Anwendungen erweiterbar. Dennoch gibt es bereits kleine Anwendungen in JAVA, wie Vokabeltrainer oder Gedächtnistrainer, die sich für kleine Übungsstunden zwischendurch eignen.

Der Favorit im Bereich der mobilen Endgeräte für mobile learning sind Smartphones. Smartphones gehören zur Gruppe der Handhelds. So bezeichnet werden tragbare Elektrogeräte, die mit einer Hand bedient werden können. Smartphones vereinigen die Funktionalität eines Mobiltelefons und eines PDA³⁶. Es ist ein Betriebssystem installiert, sodass eigene Anwendungen installiert werden können. Verfügbare Betriebssysteme sind je nach Modell Linuxdistributionen wie Android von Google, eine abgespeckte Version des Mac OS X für das iPhone, Windows Mobile von Microsoft oder BlackBerry. Außerdem verfügen Smartphones über einen Touch-sensiblen Bildschirm (Touchscreen) für die Bedienung per Finger oder mit einem speziellen Stift. Somit eignen sich diese Geräte neben den üblichen Telefondiensten, der Erledigung einfacher Bürojobs wie das Schreiben und Versenden von E-Mails, auch für die Nutzung von M-Learning-Anwendungen.

Trotz aller Euphorie bringen auch Smartphones einige Nachteile mit sich. Vorallem sind sie gegenüber Handys momentan noch relativ teuer und deshalb nicht der breiten Masse zugänglich. Außerdem ist die Einarbeitung durch den deutlich größeren Funktionsumfang aufwendiger.

Grundsätzlich kann man zusammenfassen, dass der Einsatz von Notebooks im Bereich M-Learning die wenigsten Aufwände mit sich bringt. Damit ist gemeint, dass keine neuen Anwendungen implementiert werden müssen, sondern die herkömmlichen E-Learning-Anwendungen in vollem Umfang nutzbar sind. Anders verhält sich das bei Handys und Smartphones. Schon allein wegen der viel kleineren Oberfläche, dem Display, müssen die Anwendungen angepasst werden. Außerdem sind Leistungsfähigkeit und Speicher der Endgeräte beschränkt.

Beim mobilen Lernen kann es gehäuft zu Unterbrechungen im Lernvorgang kommen, beispielsweise durch eingehende Anrufe. Es sollte deshalb möglich sein, den Lernprozess jederzeit wieder dort fortzusetzen, wo man unterbrochen würde.

³⁶ Ein PDA (Personal Digital Assistents) ist ein handlicher Kleincomputer der vorallem zur Adress- und Aufgabenverwaltung genutzt wird. Er verfügt u.a. über eine Kalender- und Notizfunktion.

M-Learning eignet sich nicht als selbstständige Lernform. Jedoch eignet es sich als Ergänzung, um Inhalte noch mal nachlesen zu können oder sogenannte „Lernhappen“ oder engl. „Knowledge Nuggets“, d.h. kleinste Lerneinheiten, zu vermitteln. Sehr geschäftige Menschen könnten Nischen- und Totenzeiten, z.B. während ihres Arbeitsweges, nutzen um zu Lernen.

3.2 Das iPhone



Abbildung 3.1: iPhone 3GS (Quelle: <http://www.apple.com/de/iphone/iphone-3gs/>)

Nach mehreren Ankündigungen und steigender Spannung bei treuen Apple Nutzern war es am 29. Juni 2007 endlich so weit. Der Verkauf des iPhone startete in den USA. Gute 5 Monate später standen die Geräte dann auch in den deutschen Läden und seit März 2008 auch in Österreich. Anfangs war der Kauf an einen 2-Jahres-Vertrag mit einem exklusiven Handyanbieter des jeweiligen Landes gebunden. In Deutschland kostete das Handy bei Markteinführung trotz Vertrag noch 399 €.

Ein Jahr nach Markteinführung, stellte Apple die zweite Generation des Gerätes in zwei Ausführungen vor, das iPhone 3G und 3GS. Das iPhone 3GS stellt eine Weiterentwicklung dar. Das S steht für Speed und weist auf die bessere Arbeitsgeschwindigkeit und

Internetverbindung hin.

Die Preise wurden relativ schnell nach der Markteinführung von Steve Job³⁷. Die Konditionen für die zweite Version waren schon wesentlich besser. Je nach Vertragsart (Abschluss für 2 Jahre zwingend erforderlich) beträgt der Preis für die 8 GB-Variante 1 bis 169,95 Euro, für die 16 GB-Version 19,95 bis 249,95 Euro.

Das iPhone verzichtet, bis auf wenige Tasten, auf manuelle Bedienelemente und bietet stattdessen ein berührungsempfindliches Display, auch als Touchscreen bezeichnet. Der 3,5 Zoll große Touchscreen ermöglicht die Gestenbedienung, die eine sehr intuitive Nutzung erlaubt. Es gibt 6 Gesten, die mit einem oder mehreren Fingern ausgeführt werden und beinahe alle Aktionen auslösen:

- Einfaches Tippen: Die unterliegende Funktion wird ausgelöst.
- Doppeltippen: Der Bildausschnitt wird um einen festen Faktor vergrößert / verkleinert.
- Wischen: Dadurch bewegt sich der Nutzer durch Listen oder Ansichten. Das Wischen ersetzt die Bedienung von Scrollleisten.
- Antippen und Halten auf Texteingabefeldern: Eine Lupenfunktion wird ausgelöst.
- Antippen und Bewegen: Dadurch kann der sichtbare Bereich innerhalb des Bildschirms verschoben werden. Einige Objekte, z.B. Listeneinträge, können mithilfe dieser Geste an eine andere Position verschoben werden.
- Spreizen und Zusammenfügen: Die wohl bekannteste Geste mit zwei Fingern beschreibt das händische Skalieren der Ansicht bzw. von Photos ect..

Neben den Tasten und dem Touchscreen besitzt das iPhone 3 Sensoren:

- einen Näherungssensor, der näherkommende Objekte erkennt, z.B. wenn das Handy ans Ohr gelegt wird, werden Eingabefunktion und Bildschirmbeleuchtung ausschaltet,
- einen Drei-Achsen-Beschleunigungssensor, der das Drehen des Gerätes durch

37 Steven „Steve“ Paul Jobs ist Mitbegründer und Vorstand der Apple Inc.

- den Nutzer erkennt und die Ausrichtung des Display automatisch anpasst, und
- einen Helligkeitssensor zur automatischen Anpassung der Display-Helligkeit an die Lichtverhältnisse.

Zur Texteingabe erscheint eine virtuelle Tastatur. Diese verfügt über eine lernfähige Worterkennung zur Autovervollständigung der Eingabe und der automatischen Korrektur von Tippfehlern. Die Größe der Display-Tastatur ist ausreichend, sodass kein spezieller Stift für dessen Bedienung nötig ist, wie bei Konkurrenzmodellen.

Der Bildschirm mit einer Auflösung von 480 x 320 Pixeln ist in den meisten Anwendungen in ähnlicher Weise gestaltet. Am oberen Rand sieht man meist eine 4 mm breite Statusleiste zur Anzeige des Akkustandes, der Zeit, des Providers und ähnlicher Informationen. Am unteren Rand befindet sich meist eine Menüleiste, z.B. um innerhalb des Projektes zwischen 2 Modi zu wechseln. Zwischen Statusleiste und Menüleiste ist das eigentliche Programmfenster positioniert.

Das iPhone verfügt über eine abgespeckte Safari-Version und Wi-Fi WLAN³⁸ und ist damit voll internetfähig. Daneben ist der Datenaustausch und die Mobilkommunikation auch über GSM und EDGE, sowie Bluetooth und mittlerweile auch UMTS.

Was jedoch fehlt, ist ein Flash-Plugin. Der fehlende Flash-Support ist sehr bedauerlich, da die meisten multimedialen, interaktiven Inhalte, vor allem kleine Spiele und Animationen, im Internet Flash-Anwendungen sind. Da es auch in absehbarer Zukunft keine Freigabe von Apple an Adobe zur Entwicklung eines Flash-Plugin für das iPhone geben wird, hat Adobe für die kommende Flash Professional-Version CS5 angekündigt, iPhone-fähige Anwendungen erstellen zu können. Die Anwendung benötigt nach dem Export keinen weiteren Player sondern läuft als Stand-Alone Anwendung auf dem iPhone. Bereits existierende Flash-Inhalte sollen sich problemlos übertragen lassen.

38 WiFi (Wireless Fidelity) ist ein Übertragungsstandard für drahtlose Funknetzwerke. Befindet man sich mit seinem Endgerät in der Nähe eines Hotspot, kann man diesen als Zugang zum Internet nutzen. WLAN (wireless local area network) bezeichnet dabei die drahtlosen lokalen Funknetzwerke. (Spiegel 2009)

Dadurch wäre Adobe nicht länger von der Gnade Apple's abhängig.
(Adobe Labs 2009)

Apple hingegen bietet allen Interessierten eine umfangreiche Entwicklungsumgebung, mit deren Hilfe man seine eigenen iPhone-Programme schreiben kann. Mehr darüber im nächsten Kapitel *3.3 Die iPhone SDK*.

Neben den bereits genannten Funktionalitäten bietet das iPhone einen Media-Player zur Wiedergabe von Audio und hoch auflösenden komprimierten Videos im H.264-Standard. Beim Abspielen eines Videos wechselt die Darstellung automatisch in die Horizontale. Das Versenden von Foto und Audio per MMS ist möglich. Seit der Betriebssystem-Version iPhone OS 3.0 können auch Videos in Textnachrichten eingefügt werden. Weiterhin gibt es eine zwei Megapixel Kamera beim 3G-Modell und eine drei Megapixel Kamera für das 3GS-Modell.

Das iPhone Modell 3G gibt es mit acht oder 16 GB, das 3GS sogar mit einem Speicher von 32 GB. Auch der Arbeitsspeicher ähnelt mit 128 MB beim 3G bzw. 256 MB beim 3GS dem eines kleinen PC.

Der Akku reicht laut Hersteller im Stand-By Modus 250 Stunden. Internet, Video und Telefonieren ist zwischen sechs und acht Stunden am Stück möglich und das Abspielen von Musik sogar 24 Stunden. Damit weißt das iPhone eine recht akzeptable Akkukapazität auf, die das Lernen von unterwegs ermöglicht, ohne, dass der Akku dann mehrmals täglich geladen werden muss.

(Wikipedia 2009e; Apple 2009)

Internetfähigkeit, großer Speicher, ein sehr übersichtliches Display, ein Player zum Abspielen von Multimediadateien und das alles handlich verpackt - das iPhone scheint das ideale Endgerät zum mobilen Lernen.

3.3 Die iPhone SDK

Mit der iPhone SDK (Software Development Kit) bietet Apple allen registrierten Nutzern die Möglichkeit sich eigene Anwendungen für ihr iPhone zu entwickeln. Dabei handelt es sich um Stand-Alone-Anwendungen. Zur Ausführung solcher Programme auf dem iPhone sind somit keine anderen Programme notwendig. Es können aber auch Webapplikationen entwickelt werden, die dann einen Browser zum Abspielen benötigen.

Die Registrierung, sowie der Download der Entwicklungsumgebung ist kostenlos und führt keinerlei Verpflichtungen mit sich. Jedoch muss man eine iPhone Developer License bei Apple beantragen, um seine Anwendungen so aufbereiten zu können, dass sie in den App Store³⁹ hochgeladen werden können. Der App Store ist ein Teil von iTunes⁴⁰, in dem die iPhone Programme zur Verfügung stehen. Mit diesem kann man sich bequem über eine mitgelieferte Applikation auf dem iPhone verbinden und gewünschte Anwendungen herunterladen. Die Anwendungen können im App Store kostenfrei, oder gegen einen beliebigen Endbetrag angeboten werden. Der Entwickler erhält 70 % der Einnahmen. Apple behält sich 30 % Provision ein. Die iPhone Developer License ist für Einzelpersonen und Unternehmen erhältlich. Sie kostet 99\$ jährlich für Einzelpersonen und Unternehmen und 299 \$ pro Jahr für Unternehmen ab 350 Mitarbeiter zur Entwicklung firmeninterner Anwendungen. Anzahl und Preis der angebotenen Anwendungen pro Entwickler unterliegen keinen Beschränkungen. Darüber hinaus gibt es für Universitäten das iPhone Developer University Program⁴¹. Damit können höhere Bildungseinrichtungen, die die iPhone Programmierung in den Stundenplan integrieren wollen, Entwicklergruppen von bis zu 200 Studenten bilden, die ihre Anwendungen direkt auf dem iPhone testen oder über eMail o.ä. austauschen können. Die Lizenz ist kostenlos, erlaubt jedoch nicht die Veröffentlichung im App Store. (iPhone Dev 2009)

39 <http://www.app-store.de/>

40 iTunes ist ein Programm, dass für Mac und PC kostenlos zur Verfügung steht. Audio und Video können abgespielt und mit iPod, iPhone und Apple TV Daten synchronisiert. Es kann sich außerdem mit einem Shop zum Erwerb von Musik u.ä. verbinden.

41 <http://developer.apple.com/iphone/program/university.html>

Die iPhone SDK⁴² beinhaltet im wesentlichen zwei Komponenten, Xcode und Cocoa Touch. Nachdem eine kurze Anleitung zur Installation gegeben wurde, werden diese im folgenden erläutert. Die iPhone SDK erfreut sich wachsender Beliebtheit bei Entwicklern und wurde seit ihrer Einführung im Juni 2008 bereit 800 000 mal heruntergeladen. (Wikipedia 2009f)

3.3.1 Installation der iPhone SDK

Wie bereits unter 3.3 *Die iPhone SDK* erwähnt, kann die Entwicklungsumgebung nach einer Registrierung kostenlos runtergeladen werden. Die neuste Version ist zu finden unter <http://developer.apple.com/technology/xcode.html>. Die Datei ist beinahe 2GByte groß. Deshalb empfiehlt es sich eine möglichst gute Breitbandverbindung zu nutzen, um die Wartezeit erträglich zu halten.

Nach dem Download kann das Installations-Paket geöffnet werden. Es beinhaltet eine Textdatei mit Informationen, das Installationsprogramm für die iPhone SDK und alle benötigten Packages. Hat man die Installation gestartet, ist nicht mehr viel Nutzerinteraktion notwendig. Der Fortlauf muss lediglich von Zeit zu Zeit mit *Continue* bestätigt werden. Darüber hinaus bekommt man eine Auswahl aller mitgelieferten Komponenten, aus denen man wählen kann, welche installiert werden sollen. Zu einem späteren Zeitpunkt kann der Speicherort ausgewählt werden.

Standardmäßig wird die Applikation nicht wie zu vermuten unter Programme abgelegt. Stattdessen wird auf gleicher Ebene ein neuer Ordner *Developer* angelegt, in dem alle Programme, die mit der SDK geliefert werden, abgelegt sind.

Ist die Installation abgeschlossen, kann die Applikation gestartet und genutzt werden.

42 <http://developer.apple.com/iphone/index.action>

3.3.2 Xcode

Xcode ist Apples Entwicklungsumgebung. Mit ihr können Programme für den Mac und jetzt auch für das iPhone entwickelt und getestet werden.

Xcode besteht im wesentlichen aus drei Werkzeugen:

- einem Source Editor,
- einem Interface Builder und
- dem iPhone Simulator.

Der Source Editor ist ein professioneller Editor zum Coden⁴³, der einige nützliche Hilfsmittel zur Einhaltung der Syntax mit sich bringt. Codevervollständigung und Syntaxhervorhebung sind neben der genauen Fehleranzeige die nützlichsten Funktionen.

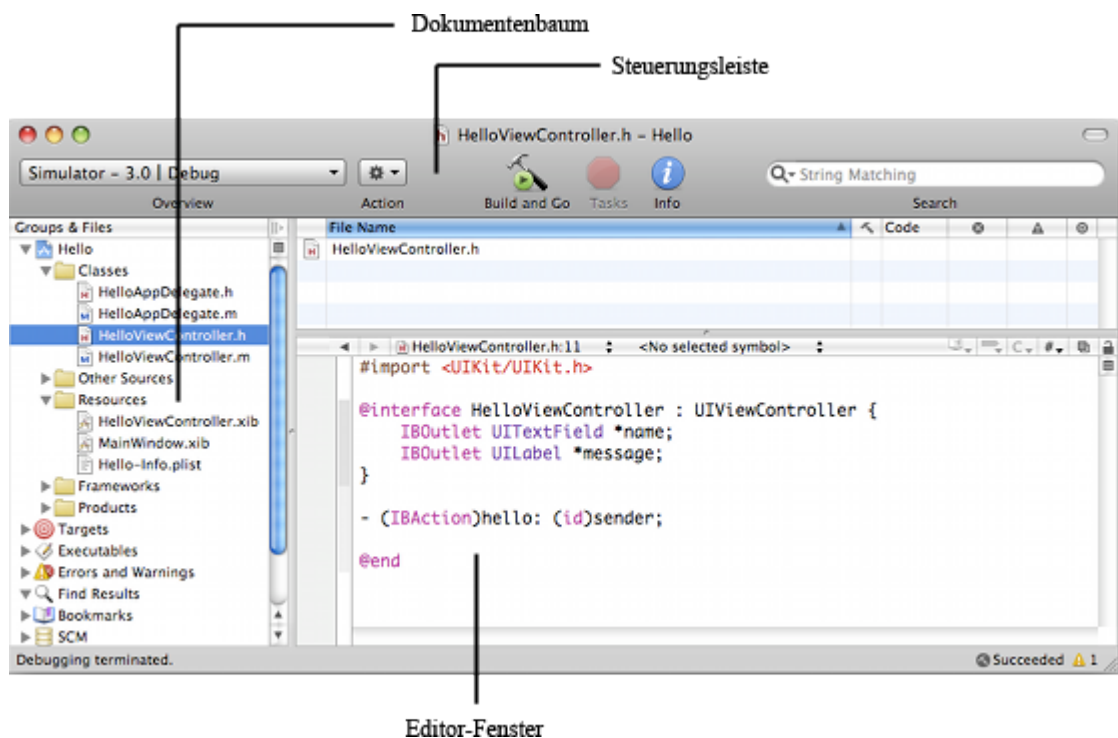


Abbildung 3.2: Programmierumgebung iPhone SDK

43 Coden ist eine gebräuchliche Kurzform für das Schreiben von Programmcode.

Die Programmoberfläche (Abb. 3.2) ist mit der anderer Programmierumgebungen zu vergleichen. Am oberen Rand ist die Steuerungsleiste mit den wichtigsten Funktionen zum schnellen Aufruf. Der wichtigste Button ist der Build & Go Button zum Testen des Programmes. Darunter links befindet sich ein Dokumenten-Baum, in dem alle Klassen, Ressourcen und sonstige Dateien der Applikation zu finden sind. Neben dem Dokumenten-Baum findet man eine Auflistung aller Dokumente, die zum aktuell ausgewählten Knoten im Dokumenten-Baum gehören. Darunter befindet sich das eigentliche Editor-Fenster, in dem die ausgewählte Datei bearbeitet werden kann.

Mit dem Interface Builder (Abb. 3.3) kann die Programmoberfläche grafisch gestaltet werden, ohne eine Zeile Code zu schreiben. Aus einer Bibliothek können vorgefertigte Elemente, wie Buttons oder Slider genutzt und nach Wünschen konfiguriert werden. So kann man sich schnell und einfach einen Prototyp seines Programmes erstellen.

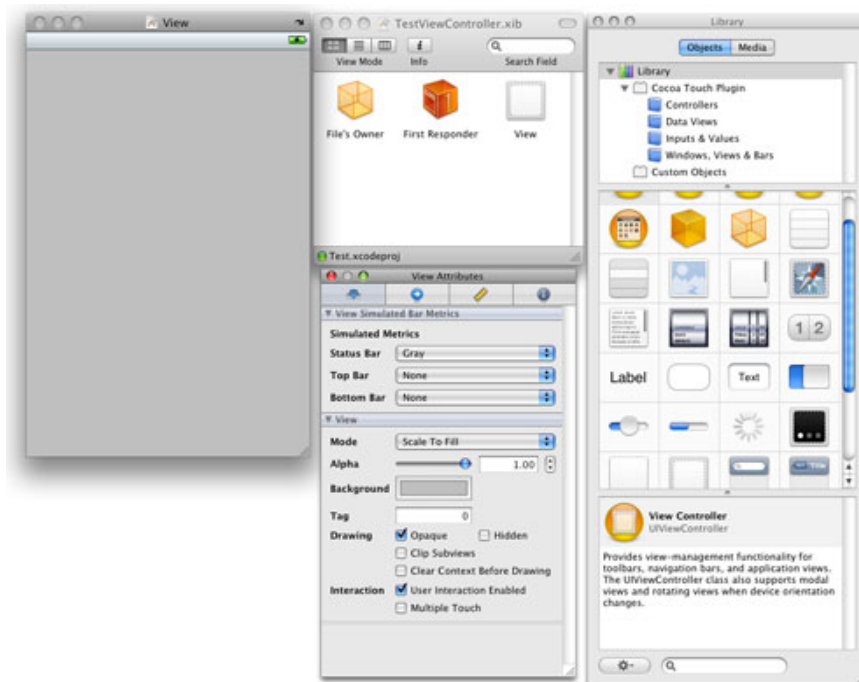


Abbildung 3.3: Interface Builder der iPhone SDK

Mit geringem Programmieraufwand wird eine Steuerung und Referenzen für die Bildschirmobjekte erstellt. Die Zuweisung erfolgt dann wieder direkt im Interface Builder über Drag & Drop. Durch diese Vorgehensweise erzeugt man schnell ein reales Programm, das nun über den Code gesteuert werden kann. Das eben erklärte Vorgehen

basiert auf dem Model-View-Controller Design. Dessen Funktionsweise wird unter 4.2 *Das Model-View-Controller Design* näher erläutert.

Zum Testen der Programme in der Entwicklungsumgebung liefert Xcode den iPhone Simulator. Er bildet das iPhone in seiner originalen Größe ab. Touch-Events werden mit der Maus simuliert.

Darüber hinaus bietet die Entwicklungsumgebung Xcode zahlreiche Instrumente um Performance und Verhalten der Anwendung messen und analysieren zu können.

Weitere nützliche Features sind zum Beispiel die *Snapshot-Funktion* und *#pragma marks*. Mittels einer Snapshot kann der aktuelle Bearbeitungsstatus gespeichert werden. Ruft man diesen zu einem späteren Zeitpunkt wieder ab, bekommt man eine Übersicht über alle Dateien in denen die Änderungen seit dem letzten Snapshot markiert sind.

Wenn ein Programm sehr lang ist, kann es schnell unübersichtlich werden. Markierungen sind sehr nützlich um Funktionen schnell zu finden. Diese werden mit dem Befehl *#pragma marks* gesetzt und sind wie alle Funktionen einer Klasse über ein Drop-Down-Menü erreichbar.

3.3.3 Cocoa Touch

Cocoa Touch ist eine ans iPhone angepasste Version der Cocoa API. Cocoa ist eine Programmierschnittstelle, mit der unter dem Betriebssystem Mac OS X von Apple programmiert werden kann.

Die Cocoa API stellt verschiedene Frameworks zur Verfügung. Als Framework wird eine Sammlung von Klassen verstanden, die zur Entwicklung von Applikationen zur Verfügung stehen und nützliche Funktionen anbieten. Es gibt unter anderem Basisklassen für die Arbeit mit Strings oder Arrays, Klassen zur Steuerung von Audio und Video und Klassen zur Behandlung der Nutzerinteraktionen.

Cocoa Touch inkludiert zusätzliche Klassen zur Auswertung von Touches, d.h. zur Behandlung von Nutzereingaben über eine berührungsempfindliche Oberfläche.

Programmiert werden kann in verschiedenen Sprachen, jedoch ist Objective-C die am häufigsten genutzte. Darüber hinaus sind alle Klassen der API in Objective-C geschrieben. Es handelt sich dabei, ähnlich wie C++, um eine auf C basierende Sprache zur objektorientierten Programmierung. Die Programmierlogik ist der von C++ sehr ähnlich, jedoch unterscheiden sich die Sprachen in der Syntax teilweise erheblich.

Beispiel:

Methodenaufruf in C++: `player.settitle(„Neues Kapitel“);`

Methodenaufruf in Objective-C: `[player settitle:@„Neues Kapitel“];`

Wie man sieht, ist die Logik die gleiche. Man ruft für ein Objekt *player* eine Methode *settitle* auf und belegt somit eine Eigenschaft des Objektes mit einem neuen Wert *Neues Kapitel*. Jedoch unterscheidet sich die Schreibweise des Befehls.

4. Umsetzung eines Media-Players für das iPhone

Die Idee ist es einen Player zur Wiedergabe von Audio-Files zu schreiben. Das allein wäre noch kein Grund, einen eignen Player zu programmieren, da das iPhone über einen Media-Player zur Wiedergabe von verschiedenen Audio- und Videoformaten verfügt. Um jedoch darüber hinaus die auditive Komponente durch visuelle Elemente zu verstärken, wird die Funktionalität des Players durch die Anzeige von Bild, Text, Animationen und Video erweitert. Die Steuerung der Wiedergabe erfolgt über die Steuerung des Audios.

Durch die Kombination von Audio, Bild, Video und Text kann Wissen audio-visuell vermittelt werden. Es werden mehrere Sinne angesprochen und der Inhalt schneller und nachhaltiger verinnerlicht.

4.1 Vorüberlegungen

Zur Wahl eines geeigneten Werkzeuges waren keine Vorüberlegungen notwendig. Zum einen war aus den Recherchen bekannt, dass das iPhone keine Flash-Unterstützung bietet und somit Adobe Flash nicht für die Entwicklung in Frage kommt. Zum anderen bietet die iPhone SDK eine sehr mächtige Entwicklungsumgebung für iPhone-Applikationen, mit der die gewünschte Funktionalität umgesetzt werden kann.

Nachdem die Funktionsweise des Players klar war, stellten sich jedoch einige Fragen zu den Medienformaten und der Datenverwaltung. Müssen Animationen und Videos als verschiedene Medien behandelt werden und wie können die wiederzugebenen Inhalte beschrieben werden, damit sie ohne Konfiguration des Players änderbar sind? Diese Fragen werden im folgenden Abschnitt erläutert.

4.1.1 Videos und Animationen

Wie bereits mehrmals erwähnt, bietet das iPhone bis zum heutigen Stand keinen Player für Flash-Inhalte. Somit ist das für Animationen häufig genutzte SWF-Format nicht abspielbar. Animationen können somit nur in den unterstützten Video-Formaten⁴⁴ eingesetzt werden. Da beide Medientypen den gleichen Formatbeschränkungen unterliegen, brauchen sie im vorliegenden Kontext nicht differenziert werden. Animationen und Videos werden unter Video zusammengefasst.

Desweiteren musste untersucht werden, ob das gleichzeitige Abspielen von Audio und Video auf separaten Playern möglich ist, ohne das Eines vom Anderen unterbrochen wird. Die iPhone SDK bietet hierfür zwei Player für Audio und Video, die sich nicht gegenseitig beeinflussen. Eine Unterbrechung ist folglich zu keiner Zeit nötig und die nahtlose Integration von Video in den Abspielvorgang problemlos möglich.

4.1.2 Die Verwaltung der Inhalte in XML-Dateien

Da der geplante Player später für verschiedene Produkte wie Hörbücher oder audiovisuelle Podcast genutzt werden soll, war zu überlegen, wie die darzustellenden Inhalte an den Player übergeben und deren Anzeige zeitlich gesteuert werden kann, ohne die Inhalte und dessen Abfolge fest in der Programmlogik implementieren zu müssen.

Dazu wird zu jedem Audio-File ein XML-File geschrieben, in dem alle notwendigen Informationen zu den Inhalten gespeichert werden, die der Player zu dessen Anzeige benötigt. Der Aufbau dieser XML-Datei wird im folgenden Abschnitt *4.1.3 Aufbau der XML-Datei* genauer erläutert. Durch die Funktionalität des Players alle notwendigen Informationen aus einer XML-Struktur zu lesen, sind die Inhalte stetig veränder- und erweiterbar, ohne den Player an sich konfigurieren zu müssen.

⁴⁴ Mehr zu den unterstützten Videoformaten ist unter *4.1.3 Auf der XML-Datei* zu finden.

Zur Demonstration der Funktionalität des Players wurde eine kleine Tour durch zwei der virtuellen Lernwelten der Firma create.at⁴⁵ entwickelt.

4.1.3 Aufbau der XML-Datei

Tabelle 4.1: Übersicht über XML-Struktur

	info	image	text	movie
time		X	X	X
url	X	X		X
mtype	X			X
duration		X		
description	X	X	X	

Die XML-Datei besteht aus einem Info-Element, und beliebig vielen Image-, Text- und Movie-Elementen. Die Tabelle bietet eine Übersicht über die Elemente und den jeweils notwendigen Tags zur Beschreibung der Inhalts-Komponenten. Im folgenden werden die Elemente näher erläutert.

info

Im innerhalb des info-tag stehen die Hauptinformationen zum aktuellen Lernabschnitt / Kapitel.

image

Unterstützt werden die gängigsten Formate .tif, .jpg, .gif und .png. Ebenfalls dargestellt werden können aber auch das Windows Icon Format (.ico), das Windows Bitmap Format (.bmp), Windows Cursor (.cur) und XWindow bitmap (.xbm).

text

Hier wäre es denkbar einen weiteren Tag image einzufügen, der angibt, ob das zuvor gezeigte Bild stehen bleiben soll oder nicht. Momentan wird weiterhin das zuletzt aufgerufene Bild angezeigt, der Text steht also zu keinem Zeitpunkt allein im

45 www.create.at

Anzeigebereich.

video

Die unterstützten Videoformate der MPMoviePlayerController Klasse sind .mov, .mp4 und .3gp. Für diese Formate sollte einer der folgenden Kompressions-Standard verwendet worden sein:

- H.264 Baseline Profil mit einer Auflösung von höchstens 640 x 480 Pixel und einer Framerate von 30 Frames pro Sekunde oder
- MPEG-4 Part 2 Standard Profil.

Das Video-File soll keine Audio-Spur enthalten, da das Hauptaudio nicht unterbrochen wird und so den akustischen Hintergrund zum Film bildet.

time

Das time-tag wird für Bilder, Texte und Videos gesetzt und beschreibt den Zeitpunkt, zu dem das Element angezeigt wird. Angegeben wird der Zeitpunkt in Sekunden.

Beispiel: <time>85</time>

Das Element wird nach **einer Minute und 25 Sekunden** nach dem Starten des Audios angezeigt.

url und mtype

Im url-tag wird der Name der Datei angegeben. Bei Bildern muss die url aus dem Namen und der Dateierweiterung bestehen. Anders ist das innerhalb des Info- oder Movie-Elementes. Hier wird nur der Name angegeben. Zusätzlich gibt es deshalb einen mtype-tag, indem die Dateierweiterung benannt wird.

Beispiel image: <url>bild.jpg</url>

Beispiel movie: <url>movie</url>
 <mtype>mov</mtype>

Es ist nicht nötig, einen kompletten Pfad zu übergeben. Dieser wird bei der Initialisierung der jeweiligen Objekte automatisch gesetzt.

duration

Das duration-tag wird derzeit ausschließlich für Texte gesetzt und beschreibt die Dauer in Sekunden, für den der Text angezeigt wird. Denkbar wäre solch ein Tag auch für Bilder. Diese werden momentan so lange gezeigt, bis ein neues Bild oder ein Video gesetzt werden.

Duration wird im selben Format angegeben wie time.

description

Dieses Tag ist für info, image und text erforderlich. Innerhalb des info-Element steht hier die Überschrift, die Bezeichnung für die aktuelle Einheit, die wiedergegeben wird. Im image- und text-Element enthält es den Text, der angezeigt werden soll.

Aber warum ein zusätzliches text-tag?

Momentan ist der Player so programmiert, dass zu einem Bild ein Text angezeigt werden kann. Ist der description-tag des image-Element leer, wird nur das Bild angezeigt. Ansonsten bleibt der Text so lange stehen, bis ein neues Bild oder Video angezeigt wird. Um nun verschiedene Texte nacheinander zu ein und demselben Bild anzeigen zu können, gibt es das text-Element. Die Bildbeschreibung wird dadurch ersetzt, während das dargestellte Bild dasselbe bleibt.

Beispiel:

```
<image>
  <time>23</time>
  <url>eb10.jpg</url>
  <description>Eure virtuelle Lernwelt ...</description>
</image>
<text>
  <time>25</time>
  <description>Schloss EVVA</description>
  <duration>3</duration>
</text>
```

23 Sekunden nach Starten des Abspielvorgangs wird das Bild `eb10.jpg` angezeigt. Dazu ist die Beschreibung „`Eurer virtuellen Lernwelt...`“ zu sehen. Nach zwei Sekunden, also

25 Sekunden nach Starten des Players wird der Text neu gesetzt. Nun ist „Schloss EVVA“ zu lesen. Das Bild bleibt das Gleiche. Nach weiteren drei Sekunden wird der Text wieder entfernt. Das Bild wird nun ohne Text angezeigt.

4.2 Das Model-View-Controller Design

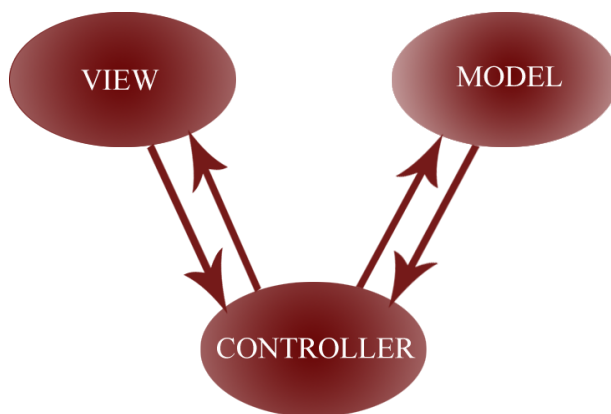


Abbildung 4.1: Model-View-Controller Architektur

Das Model-View-Controller Design ist eine bekannte Softwarearchitektur, die die strikte Trennung von Daten, Darstellung und der dahinter liegenden Programmlogik unterstützt. (Cocoa 2009)

Das Modell (engl. model) der Anwendung enthält alle Daten der Applikation. Dazu gehören auch die Klassen und andere Programmelemente zur Beschreibung dieser Daten.

Die Benutzeroberfläche der Anwendung wird als Präsentation oder engl. view bezeichnet. Dazu gehört also jeder Teil des Programmes, der für den Nutzer sichtbar ist und über den er mit dem System interagieren kann. Die Benutzeroberfläche leitet Nutzereingaben an den Controller weiter, der diese dann verarbeitet.

Der Controller (engl. Controller), oder auch die Steuerung, ist die Schnittstelle zwischen View und Model. Er verarbeitet Nutzereingaben und steuert die Anzeige der Daten aus dem Model auf der Benutzeroberfläche.

4.2.1 Model des Players

Das Modell des Players besteht aus den Bild-, Ton- und Videodateien, sowie den dazugehörigen Klassen zu dessen Beschreibung. Im vorliegenden Projekt ist das eine Klasse zum Parsen von XML-Dokumenten, in denen alle benötigte Informationen zu den darzustellenden Inhalten ausgelesen werden. Neben der XMLParser-Klasse gibt zusätzlich zu jedem Medientyp je eine Klasse zur Speicherung der jeweiligen Eigenschaften.

Trifft der XML-Parser auf ein Element im XML-File, legt er ein Objekt der Klasse des gefundenen Medientyps an und weist dieser die gefundenen Eigenschaften zu. Alle Objekte eines Typs werden dann in einem Array gesammelt um vom Controller weiter genutzt werden zu können.

Grob umschrieben werden folgende Kommandos nacheinander ausgelöst:

- *appDelegate.images = [[NSMutableArray alloc] init];*
Es wird ein Array für alle Image-Objekte angelegt und Speicher dafür freigegeben.
- *aImage = [[Image alloc] init];*
Wenn der Parser auf ein image-Tag trifft, wird ein Image-Objekt angelegt ...
- *[appDelegate.images addObject:aImage];*
... und dem Array hinzugefügt.
- *[aImage setValue:currentElementValue forKey:elementName];*
Hier werden dem Image-Objekt seine Eigenschaften zugewiesen.
- *[aImage release];*
Zum Schluss wird das Objekt und damit der Speicher wieder freigegeben.

4.2.2 View des Players

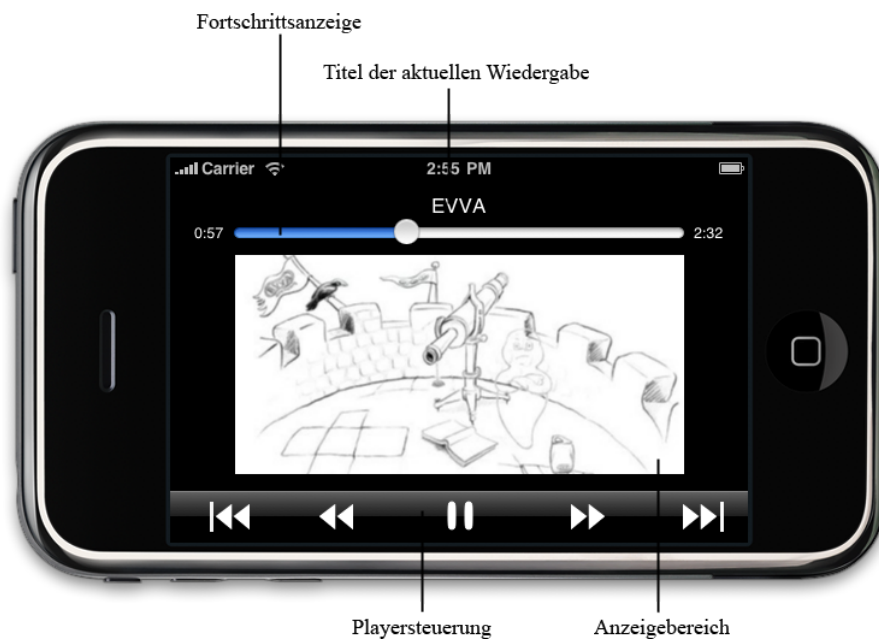


Abbildung 4.2: Benutzeroberfläche des Players

Die Nutzeroberfläche (Abb. 4.2) besteht im wesentlichen aus 3 Bereichen. Von oben nach unten sind das die Fortschrittsanzeige, der Anzeigebereich und eine Steuerungsleiste mit den Navigationselementen im unteren Bereich. Über der Fortschrittsanzeige in Form eines Sliders befindet sich darüber hinaus noch ein Label mit der Bezeichnung des aktuellen Inhaltes bzw. des Kapitels.

Die Steuerungselemente sind auf die zur wesentlichen Steuerung des Players notwendigen beschränkt. Es gibt einen Play-Button zum Starten des Abspielvorganges. Ist der Abspielvorgang aktiv, wechselt dieser in einen Pause-Button, mit dem das Abspielen unterbrochen werden kann. Außerdem gibt es vier Elemente zum Vor- und Zurückspulen. Mit $|<< \text{ bzw. } >>|$ kann man an den Anfang bzw. das Ende des Audio-Files springen. Mit $<< \text{ bzw. } >>$ kann der Nutzer innerhalb der Wiedergabe von einem zum letzten bzw. nächsten Abschnitt springen. Für die Navigationselemente wurden Symbole verwendet, die von anderen Media-Playern und auch schon von den üblichen Wiedergabegeräten bekannt sind.

Über den Slider, der Fortschrittsanzeige, ist keine Steuerung innerhalb des Audios möglich. Er zeigt lediglich den Fortschritt im Abspielvorgang an.

Der Anzeigebereich im mittleren Teil ist für die Anzeige von Bild und Text.



Abbildung 4.3: Anzeige während des Abspielens eines Videos

Filme werden in Vollbild wiedergegeben. Das heißt, der Movie-Player legt sich über die restliche Anwendung und verdeckt somit auch die Steuerung unseres Players. Damit der Nutzer trotzdem selbst darüber entscheiden kann, ob das Video für ihn interessant ist oder nicht, gibt es einen Skip-Button (Abb. 4.3) zum Überspringen des Videos.

Die Oberfläche wurde mithilfe des Interface-Builder erstellt. Für die Fortschrittsanzeige wurde ein Slider gewählt und zwei Label zur Anzeige der aktuellen Abspielzeit und der Gesamtlänge des Audio. Der Anzeigebereich besteht aus einem ImageView-Element zur Anzeige von Bildern und einem TextView zur Anzeige von mehrzeiligem Text. Dieses TextView wird sowohl für die Ausgabe der Bildbeschreibung als auch zur Anzeige der Textelemente genutzt. Für die Steuerungselemente am unteren Display-Rand ist ein Toolbar-Element mit fünf Buttons für die Navigation eingesetzt. Der Play-Button wechselt, wie oben beschrieben nach dem aktuellen Status des Players, nicht nur seine Erscheinung, sondern auch seine Funktion.

4.2.3 Controller des Players

Die Controller-Klasse beinhaltet alle Methoden zur Steuerung des Players und der möglichen Nutzereingaben. Die Methoden `(void)updateViewForPlayerState` und `(void)updateViewForPlayerInfo` steuern die Anzeige der Inhalte und die der Fortschrittsanzeige. Beide Methoden werden über ein Timer-Ereignis in regelmäßigen Abschnitten aufgerufen und so das Display ständig aktualisiert. Der Controller liest die Daten aus dem Array und weiß so, wann die Elemente angezeigt werden müssen. Ist der Zeitpunkt gekommen, weißt der Controller den View-Elementen die Eigenschaften der Objekte zu.

Beispiel: Zu jeder Sekunde überprüft der Controller, ob es einen Eintrag im Array gibt, dessen `time`-Parameter mit der aktuellen Zeit übereinstimmt. Trifft der Controller dabei auf ein Bild, übergibt der Controller dem `ImageView`, einer Interface Komponente, das darzustellende Bild und dem `TextView` die dazugehörige Beschreibung.

Darüber hinaus werden alle Berührungen der Nutzer verarbeitet. Dazu gehören die Steuerungselemente des Players, sowie die `Skip`-Funktion für Videos.

5. Zusammenfassung

Im Folgenden wird es ein Fazit zur Arbeit an der Diplomarbeit und den erarbeiteten Ergebnissen geben. Nachdem der Player voll funktionstüchtig ist, gibt es dennoch noch einige Erweiterungsvorschläge. Diese werden im zweiten Teil dieser Zusammenfassung erläutert.

5.1 Fazit der Diplomarbeit

Die Arbeit an der Diplomarbeit verlief in mehreren Phasen. Zu Beginn galt es mehr über das Konzept der virtuellen Welten zu erfahren und dieses Wissen zu strukturieren. Schon während dessen wurde es an mehreren Stellen deutlich, dass das Thema unendlich viele Betrachtungsweisen ermöglicht und diese Arbeit nicht mehr als einen Überblick bieten kann. Einzelne Bereiche, wie zum Beispiel der Abschnitt über Avatare, hätten eine eigene Arbeit füllen können. Deshalb gestaltete es sich von Zeit zu Zeit schwierig die richtige Balance zwischen zu vielen und zu wenigen Informationen zu finden.

Nachdem alle wichtigen Merkmale virtueller Lernwelten erkannt und erklärt wurden, konnte eine Typisierung der Vielzahl von Lernprogrammen vorgenommen werden. Außerdem stellte sich der Mehrwert virtueller Welten zum Lernen heraus. Sie fesseln den Nutzer, lassen ihn in eine neue Welt eintauchen und machen dadurch das Lernen zu einem Erlebnis. Außerdem werden beinahe alle Sinne des Lerners angesprochen, was zu maximalen Gehirnleistungen führt und das Lernen nachhaltig werden lässt. Darüber hinaus bieten virtuelle Welten einen Platz, um sich mit Anderen zu treffen, zum gemeinsamen Lernen oder um Rat zu bekommen. Nicht ganz klar ist, ob sie auch einen wirtschaftlichen Mehrwert bringen. Die Produktionskosten für individuelle Lernwelten sind sehr hoch. Hinzu kommen relativ hohe Ressourcenaufwände für sehr komplexe 3D-Welten. Natürlich können durch virtuelle Welten Seminarkosten etc. gespart werden, dennoch steht dessen Höhe in Relation zur Größe des Unternehmens und des Einsatzes der virtuellen Lernwelt innerhalb dieses Unternehmens.

Nebenbei wurde über eine Möglichkeit nachgedacht, virtuelle Welten auf dem iPhone zugänglich zu machen. Das iPhone wurde deshalb gewählt, da es zu den Lieblingen unter den Smartphones gehört und als solches die Voraussetzungen für mobile learning mit sich bringt. Während den Recherchen zum Thema mobiles Lernen wurde schnell klar, dass sich ganze virtuelle Welten auf einem mobilen Endgerät nicht sonderlich zum Lernen eignen. Zum einen zeichnet sich mobile learning durch extrem kurze Lernphasen aus, in denen der Lerner in einer virtuellen Welt nicht weiter kommen würde. Zum anderen würde allein das Laden einer komplexen Anwendung durch den enormen Ressourcenverbrauch zu lange für das Lernen zwischendurch dauern. Auch das im Vergleich zum PC kleine Display könnte die Navigation innerhalb virtueller Welten erschweren, weil immer nur ein sehr kleiner Ausschnitt sichtbar wäre.

Deshalb entstand die Idee eines Players, der sich zur Wiedergabe kleiner Lerneinheiten aus Bild, Text und Video zu einem Audio eignet. Nach einer Einarbeitungsphase in die Entwicklungsumgebung, der iPhone SDK, und einigen notwendigen Vorüberlegungen (*4.1 Vorüberlegungen*) konnte die Umsetzung erfolgen. Jedoch ist die Entwicklung noch nicht abgeschlossen. Welche Erweiterungen den Player leistungsfähiger und flexibler einsetzbar machen würde, wird im anschließenden Abschnitt erläutert. Trotzdem kann man von einer erfolgreichen Realisierung des Projektes sprechen, da die Grundfunktionalität des Players gegeben ist.

5.2 Erweiterung des Players

Derzeit ist es noch möglich, den Nutzer selbst darüber entscheiden zu lassen, welche Lerneinheit er sich anschauen möchte, da die XML-Datei nicht dynamisch an den Player übergeben wird, sondern in der Programmlogik fest implementiert ist. Das sollte unbedingt geändert werden, da sich der Nutzer so für jede neue Lerneinheit den gesamten Player runterladen müsste.

Deshalb ist es wünschenswert, dass es beim Start der Anwendung vorab eine Auswahl aller verfügbaren Lerneinheiten gäbe. Entscheidet sich der Lerner für eine Einheit,

werden alle benötigten Dateien (XML-File, Audio-File, Bilder und Videos) heruntergeladen und auf dem lokalen Datensystem gespeichert. Erst dann startet der Player und initialisiert sich mit dem aktuellen Inhalten.

Darüber hinaus wäre eine Zusatzfunktion zum Speichern des aktuellen Status bzw. die automatische Speicherung bei Beendigung des Programmes hilfreich, damit der Lerner bei längeren Lerneinheiten die Wiedergabe an jeder Position unterbrechen und später wieder fortsetzen kann.

Abkürzungen

3D / 2D / 1D	drei- / zwei- / eindimensional
AI	Artificial Intelligence
API	Application Programming Interface
CPU	Central Processing Unit
EDGE	Enhanced Data Rates for GSM Evolution
GSM	Global System for Mobile Communications
HTML	Hypertext Markup Language
SWF	Shockwave Flash
UMTS	Universal Mobile Telecommunications System
VRML	Virtual Reality Modeling Language
XML	Extensible Markup Language

Glossar

API	engl. für Programmierschnittstelle. Über sie hat der Nutzer Zugang zu Klassen, mit denen er seine Programme schreiben kann.
Bandbreite	Frequenzbereich des Datenübertragungssystem, in dem die Übertragung möglich ist
Backend	Schichten eines Programmes können als Frontend und Backend bezeichnet werden. Dabei ist das Backend näher am System, das Frontend näher am Nutzer.
Browser	Applikation zum Öffnen von online-Anwendungen
Chat	engl. für plaudern, elektronische Kommunikation, meist in Textform, über das Internet
CPU	Central Processing Unit, auch Hauptprozessor oder einfach Prozessor, ist die zentrale Verarbeitungseinheit im Computer.
Client	Ein Client ist eine Anwendung auf Nutzerseite und kann einen Dienst bei Server anfragen, der diesen Dienst bereitstellen kann.
Controller	engl. für Steuerung oder Steuerungseinheit
Drop-Down-Menü	Auflistung von Menüpunkten, die bei Mauskontakt aufklappen

EDGE	Technik zur Erhöhung der Übertragungsrate in GSM-Mobilfunknetzen
Forum	virtueller Platz im Internet zum Austauschen und Archivieren von Gedanken
GSM, UMTS	GSM und UMTS sind Mobilfunkstandards erster und zweiter Generation zur Datenübertragung
HTML	Beschreibungssprache für Seiten im Internet zur Strukturierung von Inhalten
Implementation	Programmierung, Schreiben von Applikationen
Interface	Benutzeroberfläche, alle Teile einer Anwendung, die für den Nutzer sichtbar sind
JAVA	objektorientierte Programmiersprache
JAVA-Applet	Applikation, die in JAVA geschrieben ist und normalerweise in einem Browser gestartet wird
Know-How	spezifisches Wissen
Kompressions-Standard	Standard zur Komprimierung von Daten um Speicher zu sparen
online-Anwendungen	Applikationen, die in einem Browser gestartet werden können, ohne dass Teile auf dem lokalen Datensystem installiert werden müssen

Open Source	frei erhältliche und zugängliche Software, dessen Quelltext öffentlich und erweiterbar ist
Plug-in	Software, die sich an ein anderes Softwareprodukt anschließt und so dessen Funktionsumfang erweitert
Render-Engine	Programmkomponente zum Berechnen der grafischen Objekte einer Darstellung
SDK	engl. für Entwicklungsumgebung, stellt alle Werkzeuge bereit, die ein Entwickler zum Schreiben von Anwendungen braucht.
Server	Kommuniziert mit Client um diesem Dienste zugänglich zu machen
Skript	in der Informatik Schriftsatz zur Beschreibung eines Algorithmus
Stand-Alone-Prog.	Applikationen, die kein Programm zu dessen Ausführung benötigen
Troublemaker	engl. für „Ärgermacher“, Person, die Unruhe stiftet
VoiceOperIP	Internettelefonie, Telefonieren über Computernetzwerke
VRML	Format zur Beschreibung von 3D-Szenen
XML	Sprache zur Strukturierung von Daten in Textform
XML-Parser	Programm zum durchsuchen und aulesen einer XML-Datei

Abbildungs- und Tabellenverzeichnis

Abbildung 2.1: Techforce	19
Abbildung 2.2: WhizLow	21
Abbildung 2.3: Babus Hold	26
Abbildung 2.4: Steve	30
Abbildung 2.5: PPP Persona	31
Abbildung 2.6: Adele	32
Abbildung 2.7: Verhaltensgenerierung Herman the Bug	37
Abbildung 2.8: links: RescuSim, rechts: INMEDIA	52
Abbildung 2.9: MiRTLE Projekt	54
Abbildung 2.10: Scoyo	56
Abbildung 3.1: iPhone 3GS	63
Abbildung 3.2: Programmierumgebung iPhone SDK	69
Abbildung 3.3: Interface Builder der iPhone SDK	70
Abbildung 4.1: Model-View-Controller Architektur	78
Abbildung 4.2: Benutzeroberfläche des Players	80
Abbildung 4.3: Anzeige während des Abspielens eines Videos	81
Tabelle 1.1: Überblick über die Lernparadigmen	10
Tabelle 4.1: Übersicht über XML-Struktur	75

Literaturverzeichnis

(Adobe Labs 2009)

URL: http://labs.adobe.com/wiki/index.php/Applications_for_iPhone (Stand 10.11.09)

(Anderson 2008)

Anderson, Fritz: *Xcode 3 Unleashed*. Addison-Wesley Longman. Amsterdam 2008

(Apple 2009)

URL: <http://www.apple.com/iphone/> (Stand 06.11.09)

(Baumgartner, Payr 1994)

Baumgartner, Peter; Payr, Sabine: *Lernen mit Software*. Österreichischer Studien Verlag. Innsbruck 1994

(Baumgartner, Payr 1994b)

Baumgartner, Peter; Payr, Sabine: *Wie Lernen am Computer funktioniert*. in: c't 8/94. S. 138-142. Verlag Heinz Heise. Hannover 1994

(Bendel 2003)

Bendel, Oliver: *Virtuelle Tutoren*. In: bulletin. 4/2003. S. 34 - 36

(Bendel 2003b)

Bendel, Oliver: *Pädagogische Agenten im Corporate E-Learning*. Dissertation. St. Gallen 2003

(Chen & Michael 2006)

Chen S., Michael D.: *Serious Games: Games That Educate, Train and Inform*. Course Technology PTR . 2006

(Cocoa 2009)

URL: Quelle: www.cocoa-coding.de/mvc/mvc.html (Stand 10.10.09)

(DeFreitas 2008)

DeFreitas, Sara: *Serious Virtual Worlds: a scoping study*.

URL: <http://www.jisc.ac.uk/media/documents/publications/seriousvirtualworldsv1.pdf>
(Stand 23.07.09). 2007

(Gartner 2007)

URL: <http://www.gartner.com/it/page.jsp?id=503861> (Stand 12.09.09). Stamford, Connecticut. 2007

(Giesecke 2001)

Giesecke, Hermann: *Einführung in die Pädagogik*. Juventa. Weinheim, München 2001

(Grillhofer 2009)

Grillhofer, Peter: *Edutainment – Spiele im Gesundheitswesen*.

URL: http://de.wikibooks.org/wiki/Edutainment_-_Spiele_im_Gesundheitswesen
(Stand 28.07.09)

(Haag 1995)

Haag, Martin: *Gesamtkonzept für die Entwicklung und den Einsatz von computerunterstützten Lehr-/Lernsystemen in der Mediziner Ausbildung an der Universität Heidelberg*. Diplomarbeit an der Universität Heidelberg/Fachhochschule Heilbronn 1995

(Ilmenau 2009)

URL: <http://www4.tu-ilmenau.de/m-learning/> (Stand 02.10.09)

(iPhone Dev 2009)

URL: <http://developer.apple.com/iphone/program/> (Stand 06.11.09)

(Kelly 2009)

Kelly, Henry: *FAS Virtual Worlds Whitepaper*.

URL: http://vworld.fas.org/wiki/Virtual_Worlds_Whitepaper (Stand 01.10.09)

(Kochan 2009)

Kochan, Stephan G.: *Programming in Objective-C 2.0: A complete introduction to the Objective-C language for Mac OS X and iPhone development*. Addison-Wesley Longman. Amsterdam 2009 (2. Aufl.)

(Krcmar & Koppenhöfer 1998)

Krcmar, Helmut; Koppenhöfer, Christine: *Multimedia in der Weiterbildung*.

URL:

[http://www.winfobase.de/lehrstuhl/publikat.nsf/intern01/0EA66866F3C5B6E8412566500029C4B5/\\$FILE/98-02.pdf](http://www.winfobase.de/lehrstuhl/publikat.nsf/intern01/0EA66866F3C5B6E8412566500029C4B5/$FILE/98-02.pdf) (Stand 07.05.09)

(Krause 2008)

Krause, Dirk: *Serious Games – The state of the game. Pixelpark Agentur Whitepaper*. Köln 2008.

(Lester et al. 2000)

Lester, James C.; Johnson, W. Lewis; Rickel, Jeff W.: *Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments*. In: International Journal of Artificial Intelligence in Education (2000) 11. S.47-78

(LexiROM 1999)

Microsoft & Brockhaus . LexiROM 4 Edition 2000. Microsoft: Unterschleißheim 1999

(Mader & Stöckl 1999)

Mader, Günter; Stöckl, Walter: *Virtuelles Lernen: Begriffsbestimmung und aktuelle empirische Befunde*. Studien-Verlag. Innsbruck 1999

(Mark & LaMarche 2009)

Mark, Dave; LaMarche, Jeff: *Beginning iPhone 3 Development: Exploring the iPhone SDK*. Apress. New York 2008

(Meyers Lexikon 1997)

Meyers Lexikon in drei Bänden. Mannheim. Bibliographisches Institut & F.A. Brockhaus AG 1997

(Schanda 1995)

Schanda F.: *Computer-Lernprogramme*. Beltz. 1995

(Schulmeister 2005)

Schulmeister, Rolf: *Interaktivität in Multimedia-Anwendungen*. URL: <http://www.e-teaching.org/didaktik/gestaltung/interaktiv/InteraktivitaetSchulmeister.pdf>
(Stand 05.08.09). 2005

(Schwan & Buder 2006)

Schwan, Stephan; Buder, Jürgen: *Virtuelle Realität und E-Learning*.
URL: <http://www.e-teaching.org/didaktik/gestaltung/vr/vr.pdf> (Stand 04.08.09). 2006

(Spiegel 2009)

URL: www.spiegel.de/netzwelt/tech/0,1518,239795,00.html (Stand 16.10.09)

(Spierling 2005)

Spierling, Ulrike: *Learning with Digital Agents – Integration of Simulations, Games, and Storytelling*. In: Burmester, M., Gerhard, D., Thissen, F. (eds.): Digital game based learning. Proceedings of the 4th International Symposium for Information Design, 2nd of June 2005 at Stuttgart Media University. S. 115-147. Universitätsverlag Karlsruhe. Karlsruhe 2006

(Thorndike 1898)

Thorndike, E. L.: *Animal Intelligence: An Experimental Study of the Associative Processes in Animals* (Psychological Review, Monograph Supplements, No. 8). Macmillan. New York 1898

(Welsch 2000)

Welsch, Wolfgang. «*Virtual to Begin with?*» *Subjektivität und Öffentlichkeit: Kulturwissenschaftliche Grundlagenprobleme virtueller Welten*. Hrsg. von Sandbothe, Mike; Marotzki, Winfried. Köln: Harlem 2000

(Wikipedia 2009a)

URL: <http://de.wikipedia.org/wiki/Virtualit%C3%A4t> (Stand 10.09.09)

(Wikipedia 2009b)

URL: <http://de.wikipedia.org/wiki/ELIZA> (Stand 09.09.09)

(Wikipedia 2009c)

<http://de.wikipedia.org/wiki/Hypermedia> (Stand 09.11.09)

(Wikipedia 2009d)

<http://de.wikipedia.org/wiki/Webinar> (Stand 02.10.09)

(Wikipedia 2009e)

URL: <http://de.wikipedia.org/wiki/IPhone> (Stand 06.11.09)

(Wikipedia 2009f)

URL: http://de.wikipedia.org/wiki/App_Store#iPhone_SDK (Stand 06.11.09)

(Will 2000)

Will, Matthias O.: *Lehr- und Lernparadigmen als Grundlage lernförderlicher Mediennutzung*. URL: <http://ad.informatik.uni-freiburg.de/mitarbeiter/will/slides/os1200.pdf> (Stand 10.09.09). 2000

(Winn 1993)

Winn, W.: *A conceptual basis for educational applications of virtual reality*. Seattle: University of Washington, Human Interface Technology Laboratory of the Washington Technology Center. Technical Publication R-93-9. 1993.

(Zimbardo & Gerrig 1999)

Zimbardo, Philip G. & Gerrig, Richard J.: *Psychologie*. Springer. Berlin, Heidelberg 1999

(Zimbardo 1992)

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

Anhang I: Sammlung virtueller Welten

NASA 3D-Welt



- <http://www.nasa.gov/city>
- 2 aufwendig animierte 3D Welten Home und City
- Präsentation von NASA-Lösungen zur Unterstützung im Alltag zu Haus (Home) und in anderen alltäglichen Situationen (City, z.B. Sport, Medizin)
- Navigation durch 3D Welt möglich, Videosequenzen integriert zur Darstellung der Entwicklungen / Produkte

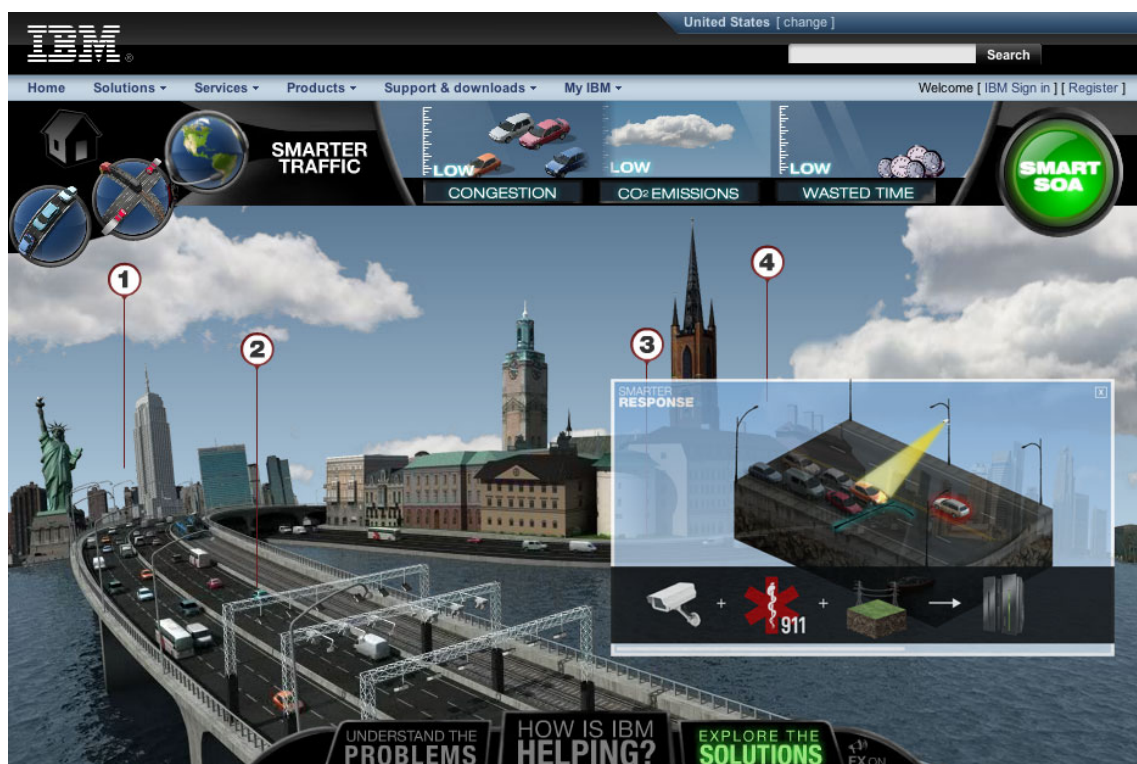
IBM – Innov8 3.0



- <http://www-01.ibm.com/software/solutions/soa/innov8/smartwork.html>

- Training für angehende Manager
- Simulator um zu zeigen wie, mittels IBM-Lösungen, Prozesse effizienter, profitabler und umweltfreundlicher gestaltet werden können
- Beispielszenarien, anhand von Nutzereingaben werden die Folgen des Handelns simuliert

IBM 3D-Welten



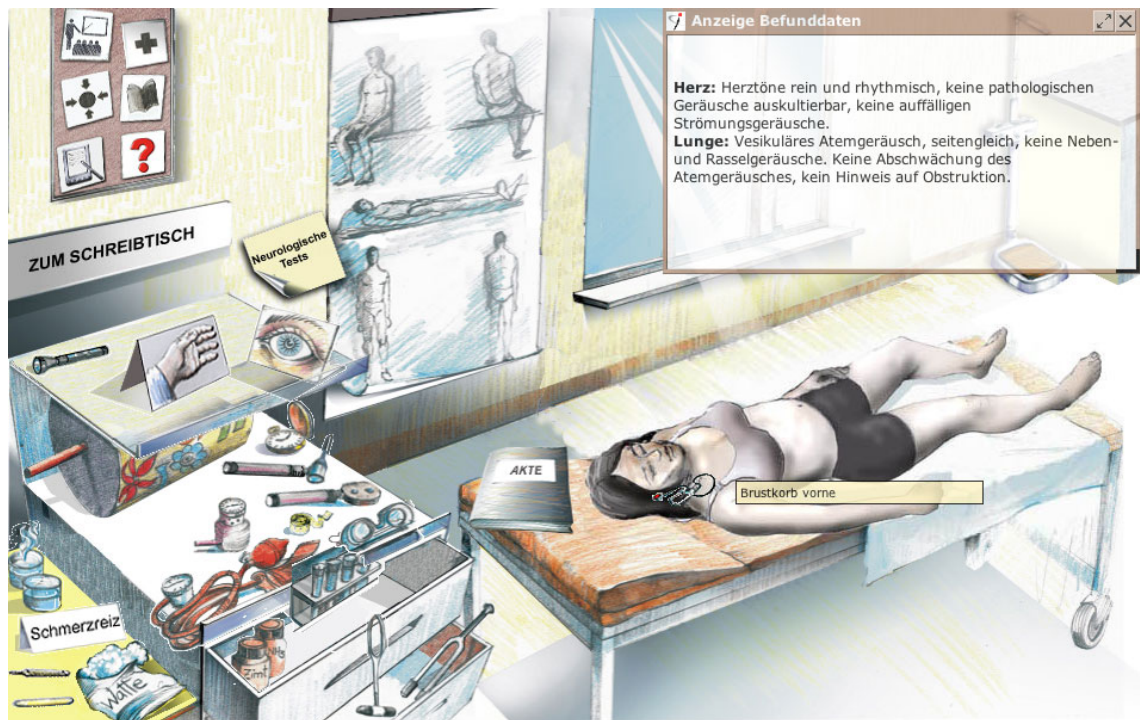
- <http://www-01.ibm.com/software/solutions/soa/innov8/smartwork.html>
- 3 aufwendig animierte 3D Welten
- IBM-Lösungen zur Verbesserung von Logistik, Kundenbetreuung und Verkehr werden simuliert

VAH – Virtuelles Autohaus



- <http://www.vah-projekt.org/>
- Gemeinschaftsprojekt des Fraunhofer Institut für graphische Datenverarbeitung in Darmstadt, der IG Metall in Frankfurt/Main und des Zentralverband Deutsches Kraftfahrzeuggewerbe e.V. in Bonn
- zur Berufsausbildung und Weiterbildung im Kfz-Gewerbe
- virtuelles Autohaus, in dem praxisnahe Vorgänge durchlaufen / trainiert werden können

INMEDIA – Fall Simulator



- <http://www.inmedea-simulator.net/>
- INMEDIA GmbH, Reutlingen (D)
- für Aus- und Weiterbildung in medizinischen Berufen
- realitätsnahe Kliniksimulation mit typischen Patienten - Untersuchen, Diagnose stellen, Therapieren

CAMPUS – Pädiatrie



- <http://www.medizinische-fakultaet-hd.uni-heidelberg.de/Beispiel-Faelle.109993.0.html>
- Medizinische Fakultät Heidelberg, für medizinische Aus- und Weiterbildung, Untersuchen, Diagnose stellen, Therapieren
- 2D, Auswahl der einzelnen Untersuchungsschritte durch Auswahl hervorgehobener Bereiche im Zimmer

RescueSim



- <http://www.rescuesim.com/>
- Vstep, Rotterdam (NL)
- interaktive 3D Trainings- / Simulationsumgebung zum Nachstellen und Proben von Notfalleinsätzen

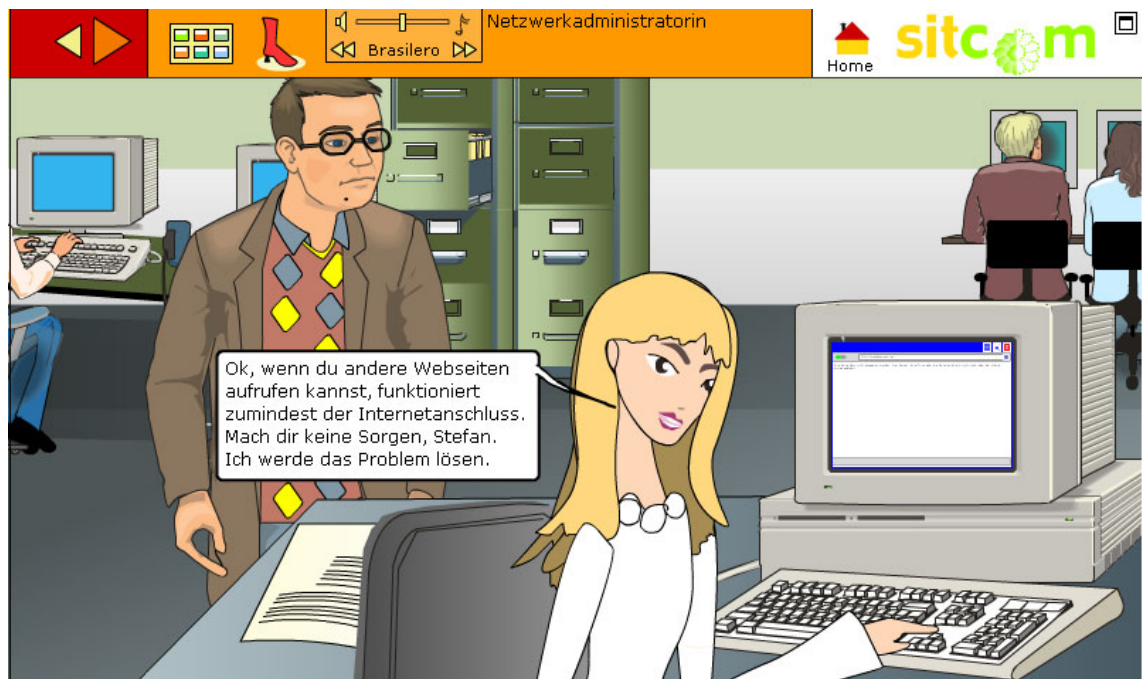
SCOYO – Lernplattform für Schüler



- <http://www.scoyo.de/>

- Scoyo GmbH in Hamburg, Tochterfirma der Bertelsmann AG
- Lernumgebung für Schüler 1.-7. Klasse
- Wissensvermittlung und –abfragen in animierten Lerngeschichten verpackt

SITCOM – „Simulating IT-Careers for wOMen“



- <http://www.sitcom-project.eu>
- Webducation Software Planungs- und EntwicklungsgmbH, Wien (Content Coordination, Drehbucherstellung, Interfacegestaltung)
- Informationen über Frauen in technischen Berufen, für Mädchen bzw. junge Frauen
- 2D, Wissensvermittlung über Dialoge zwischen Avataren, kleine Quiz / Spiele eingebaut

Techforce



- GESAMTMETALL, Gesamtverband der Arbeitgeberverbände der Metall- und Elektroindustrie e.V., Berlin
- für Jugendliche, die Ausbildung in Metall- oder Elektroindustrie anstreben -> Simulation berufsnaher Aufgaben
- Spieler (durch Avatar repräsentiert) wird Teil eines Entwicklerteams in 3D-Welt
- neben Wissenserwerb wird dieses innerhalb der Entwicklung eines fiktiven Fahrzeuges angewendet

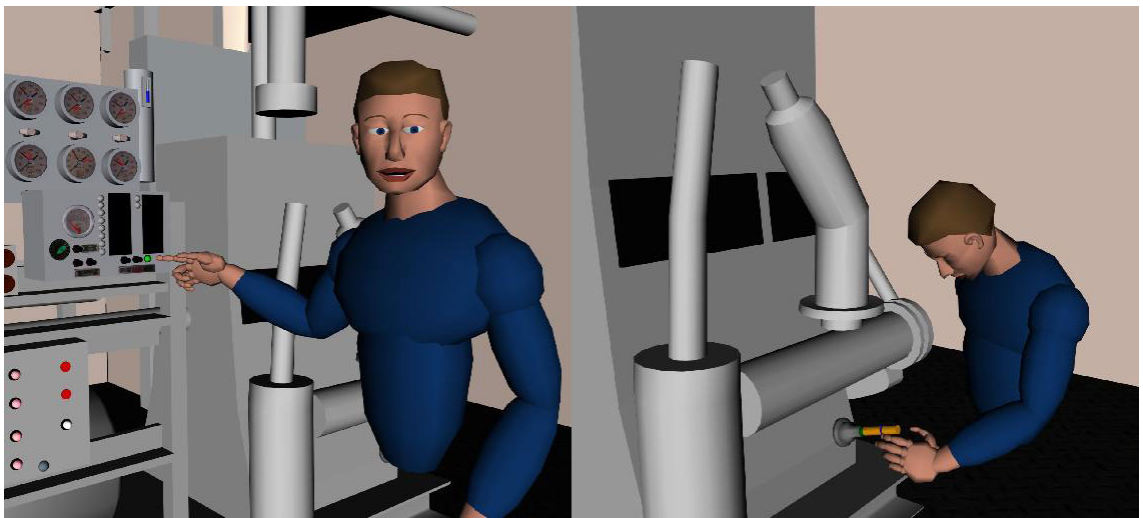
MPK20 -Sun's virtual workspace



- <http://research.sun.com/projects/mc/mpk20.html>

- 3D Umgebung, die mit Projekt Wonderland Toolkit erstellt, einem Werkzeug von Sun Microsystems um eigene 3d-Welten zu erstellen, 100% Open Source, 100% Java
- Sun Mitarbeiter können sich hier Treffen, Besprechungen abhalten, Ihre Arbeit anderen zeigen und gemeinsam an Projekten arbeiten, indem sie Anwendungen gemeinsam nutzen

Steve



- <http://www.isi.edu/isd/VET/steve-demo.html>
- CARTE (Center for Advanced Research in Technology for Education), University of Southern California
- zu Trainingszwecken der US Navy
- 3D-Umgebung, Nachbildung eines U-Boot
- Steve – interaktiver Avatar, der Lerner durch Umgebung führt, Funktionen erklärt und simuliert

Club Penguin



- <http://www.clubpenguin.com/>
- Disney Online Studios Canada
- Erlebniswelt für Kinder zum Spielen und Chatten
- 2D, aufwendig animiert

Quest for the Code



- <http://asthma.starlightprograms.org/>
- für Kinder mit Asthma, wichtige Infos über schädliche Einflüsse, wie diese zu vermeiden sind, ...
- 2D, Videosequenzen mit Erklärungen, Animationen, Spiele/Interaktionen mit sofortigem Feedback

Babus Hold & Maya Eventyr



- <http://www.mayaseventyr.dk/>
- <http://www.babushold.dk/>
- Danish Broadcasting Corporation & Danida, entwickelt von Serious Games Interactive, Copenhagen (DK)
- Aufklärungsarbeit für Kinder, unter welchen Umständen und mit welchen Schwierigkeiten ihresgleichen in Guatemala und Uganda leben
- 2D Umgebung, kleine Videos, in denen Gespräche simuliert und wichtige Infos vermittelt werden
- kleine Aufgaben, um weiter zu kommen

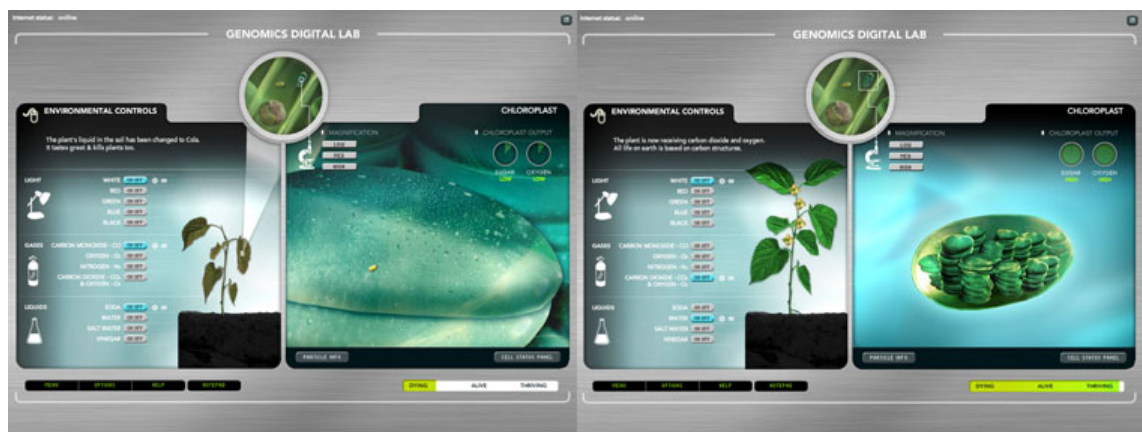
Metaplace



- <http://www.metaplace.com/>

- 3D Umgebung
- eigenes Stück Land bebauen, auf eigenem Planeten
- Content anhand einer Auswahl selbst zusammenstellen, d.h. Haus, Wände, Boden Möbel etc. nach Belieben wählen und positionieren
- andere Welten besuchen und erkunden
- mit anderen Leuten kommunizieren (Soziale Welt)

Genomics Digital Lab



- <http://www.genomicsdigitallab.com/>
- Spongelab Interactive, Toronto (Ontario, Canada)
- Lernwelt über Pflanzen, für Schüler bzw. Schulen
- Simulationen und kleine Spiele
- direktes Feedback durch Simulation der Folgen des Handelns des Nutzers

Kenyan Sign Language



- <http://www.peacecorps.gov/index.cfm?shell=learn.whatlike.interactivefeatures.ksl>
- PEACE CORPS Worldwide Schools, Washington
- Erlernen der Zeichensprache (Tiere und Früchte)
- 2 Umgebungen und 1 Spiel, kleine Animationen
- integrierte Videosequenzen, die Übersetzung in Zeichensprache zeigen

EVVA – Lernwelt



- www.create.at
- ein Produkt der create mediadesign GmbH, Wien
- Lernwelt für Mitarbeiter des Unternehmens EVVA
- Geist als Avatar, Schloss als Methaper für EVVA,
- Illustration, animiert

Selbstständigkeitserklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Wien, 17. November 2009

.....

Ellen Krüger

