

ZUKUNFT DES LERNENS –

vom Computer-Based-Training zum Web-Based-Training

Fachbereichsarbeit im Wahlpflichtfach Informatik

Ruben Merlin Mörth

Betreuer:

Mag. Carl Metnitz

GRG I Stubenbastei

Stubenbastei 6-8, 1010 Wien

Schuljahr 2005/06

Wien, im Februar 2006

Inhaltsverzeichnis

1	Einleitung	4
2	Lerntheorien	5
2.1	Behaviorismus	5
2.1.1	Umsetzung bei Lernsoftware	5
2.1.2	Kritik.....	5
2.2	Kognitivismus.....	6
2.2.1	Umsetzung bei Lernsoftware	6
2.2.2	Kritik.....	7
2.3	Konstruktivismus.....	7
2.3.1	Umsetzung bei Lernsoftware	7
2.3.2	Kritik.....	8
3	Überblick über Softwaretypen	9
3.1	Präsentations- und Visualisierungssoftware	9
3.2	Drill-and-Practice-Programme	9
3.3	Tutorielle Systeme	10
3.4	Intelligente tutorielle Systeme (ITS)	10
3.5	Simulationsprogramme	11
3.6	Mikrowelten und Modellerstellung.....	12
3.7	Hypermedia-Programme.....	12
3.8	Lernspiele	12
4	Aspekte der Softwareerstellung	14
4.1	Entwicklungsprozess	14
4.2	Konzipierung.....	16
4.2.1	Zielgruppe.....	16
4.2.1.1	Soziodemographische Daten.....	16
4.2.1.2	Vorwissen und relevante Erfahrungen potentieller User.....	16
4.2.1.3	Lerngeschichte der Zielgruppe	17
4.2.1.4	Motivation und Einstellung zum Inhalt	17
4.2.2	Lernziele und Lerninhalte	17
4.3	Usability und Software-Ergonomie.....	17
5	Die Geschichte des multimedialen Lernens und die Zukunft des E-Learnings	19
5.1	Die Zeit vor der Erfindung des PCs.....	19
5.2	Die ersten Computer.....	20
5.2.1	TICCIT	20
5.2.2	PLATO	21
5.3	Von fest-verdrahteten Lösungen zu mehr Flexibilität: Hypertext.....	21
5.3.1	NLS/Augment	22
5.3.2	XANADU.....	22
5.3.3	Hypertext Editing System (HES).....	22

5.4	Lernsoftware unter dem Einfluss des Kognitivismus.....	23
5.4.1	Adaptive generative Systeme	23
5.4.2	Intelligente tutorielle Systeme.....	23
5.4.3	Dynabook	24
5.5	Lernsoftware unter dem Einfluss des Konstruktivismus.....	24
5.5.1	Mikrowelten	24
5.5.2	Erste Hypertextsysteme.....	25
5.6	World Wide Web	26
5.6.1	Die Anfänge des Internets	26
5.6.2	WWW	26
5.6.3	Web-Based-Training (WBT)	28
5.6.4	Distance Learning: Virtuelle Universitäten	29
6	Unterrichtsorganisation mit Hilfe von Content Management Systems	31
6.1	Course Management Systems (CMS)	31
6.1.1	Moodle	32
6.1.1.1	Allgemeines	33
6.1.1.2	Technische Eckdaten	33
6.1.1.3	Open Source	33
6.1.1.4	Lerntheoretische Grundlage	34
6.1.1.5	Community	34
6.1.1.6	Die Erstellung eines Kurses in <i>Moodle</i>	35
6.1.2	Projekt <i>Aims</i>	36
6.1.2.1	Astrologieprojekt 2005.....	37
6.1.2.2	Lateinprojekt	37
7	Schlussbetrachtung: Die Zukunft des Lernens?.....	39
	Glossar	40
	Literaturverzeichnis	41

1 Einleitung

Der Gebrauch neuer Medien und Computer wird das Lernen im traditionellen Sinn auch in Zukunft nicht ersetzen können, jedoch können bei entsprechender Kenntnis die Grundvoraussetzungen – wie Motivation und Aufmerksamkeit – durch eine entsprechende Didaktik verbessert werden.

Im Zuge dieser Arbeit möchte ich eine kleine Einführung in das umfangreiche Thema der Lernsoftware-Erstellung und deren Anwendung geben.

Anfangs wird es notwendig sein, die theoretischen Aspekte des Lernens kurz zu beleuchten. Hierbei werde ich zuerst die Hauptlerntheorien vorstellen und auch analysieren, wie und wo sie im digitalen Bereich zum Einsatz kommen.

Im folgenden Kapitel soll auf die verschiedenen Softwarelösungen eingegangen werden, die bei der Umsetzung von Projekten zur Auswahl stehen.

Nach diesen beiden eher theoretischen Kapiteln werde ich ein eher praxisorientiertes Thema ansprechen. So behandelt dieser Teil meiner Arbeit die Herangehensweise an ein Projekt, in dem die Konzipierung einer bestimmten Software vorgestellt wird und mögliche Entwicklungsprozesse besprochen werden.

Der vierte Themenkomplex behandelt die Geschichte des so genannten E-Learnings. Das letzte Kapitel geht dann auf die Internet-Lernplattform *Moodle* ein, die man als eine der fortschrittlichsten und benutzerfreundlichsten unserer Zeit bezeichnen kann. In diesem Abschnitt möchte ich aber nicht nur den theoretischen Aspekten Platz einräumen, sondern auch konkrete Projekte, die mit dieser Lernplattform an meiner Schule durchgeführt werden, vorstellen, sowie Stellungnahmen von Lehrern dazu einbringen.

Da E-Learning und besonders das Internet einem ständigen Wandel unterzogen sind, können nur Teilaspekte im Bereich des multimedialen Lernens näher beleuchtet werden. Auch sei darauf hingewiesen, dass es gerade in diesem Bereich rapide Entwicklungen und immer wieder lernpsychologisch neue Erkenntnisse gibt. Deshalb mussten auch aus Rücksicht auf den Umfang dieser Arbeit mehrere interessant erscheinende Themen ausgespart bleiben.

2 Lerntheorien

Bei der Erstellung von Lernsoftware sollte der Entwickler die Lerntheorie vor Augen haben, auf der seine Software basiert und deren Vorteile er sich zu nutzen macht, um den Benutzer der Software schnellstmöglich zum gewünschten Ziel zu führen. In unserem Fall ist dies die Wissensaneignung oder das Antrainieren von bestimmten Lerneigenschaften.

Man kann zwischen folgenden Hauptströmungen von Lerntheorien unterscheiden:

2.1 *Behaviorismus*

Der Behaviorismus entstand Anfang des 20. Jahrhunderts. Unter anderem von Iwan Pawlow geprägt wurde die Theorie der klassischen Konditionierung. Pawlow verband in seinen Experimenten den Erhalt von Nahrung mit einem zuvor ausgeführten Reiz. Nach einigen Malen assoziierte das Versuchstier den Reiz mit Essen und bekam schon vor dem Erhalt der Nahrung Speichelfluss. Vertreter des Behaviorismus gehen davon aus, dass Lernen immer auf einem Reiz-Reaktionsmuster beruht. Auf dieses Prinzip stützt sich beispielweise das Vokabellernen.

Der Lernvorgang im Behaviorismus ist einseitig durch den Lehrenden geprägt, und kann vom Lernenden nicht beeinflusst werden. Dieser spielt nur eine passive Rolle, indem er eine gewisse Menge des Lernstoffs in einer bestimmten Zeit, ohne Rücksicht auf vorhandenes Vorwissen, vermittelt bekommt. Innere Vorgänge der Lernenden, wie etwa Emotionen oder Lern- und Verständnisschwierigkeiten, werden außer Acht gelassen. Im Alltag ist uns diese Art des Unterrichtens als Frontalunterricht bekannt.

2.1.1 Umsetzung bei Lernsoftware

Besonders bei älterer Lernsoftware war diese Lerntheorie sehr präsent, da die Programmierer noch nicht die technischen Möglichkeiten der Umsetzung zur Verfügung hatten, die sie heute haben. Auch primitive Vokabeltrainer, wie sie oft im Internet anzutreffen sind, stützen sich fast ausschließlich auf diese Lerntheorie.

2.1.2 Kritik

Die einseitige Vermittlung von Wissen ist normalerweise nur bei Weitergabe einfacherer Lerninhalte sinnvoll, wie es vergleichsweise beim konditionierten Lernen zu sehen ist. Im Behaviorismus kann nur Faktenwissen ohne Bezug auf die

Anwendung vermittelt werden, was in vielen Wissensbereichen nicht zielführend ist. Bei Skinners Lernmaschinentheorie¹ zeigt sich, dass der Lernstoff 1:1 wiedergegeben werden sollte, was das Verständnis bisweilen noch mehr einschränkt. Das Auslösen einer bestimmten Reaktion durch einen bestimmten Reiz ist jedoch oftmals unverzichtbar, wie zum Beispiel in Arbeitsgebieten, wo Routinearbeit ausgeführt und schnell auf Veränderungen reagiert werden muss.

2.2 Kognitivismus

„Ziel kognitivistischen Lernens ist die Förderung des Konzeptlernens und die Ausbildung von Problemlösungsfähigkeiten“²

Diese Lerntheorie hat ihren Ursprung in den 20er Jahren des 20. Jahrhunderts. Im Gegensatz zum Behaviorismus, bei dem auf einen gewissen Reiz eine bestimmte Antwort zu erfolgen hat, steht beim Kognitivismus die Problemlösung im Mittelpunkt. Diesem Handeln geht das aktive Aneignen von Fähigkeiten voraus. Man bekommt den Lernstoff vermittelt und filtert sich dabei individuell heraus, was einem persönlich am wichtigsten erscheint. Im Gehirn erfolgt dann die Informationsverarbeitung.

In den 60er Jahren des 20. Jahrhunderts entwickelte Bruner³ die „*Theorie des entdeckenden Lernens*“. Hierbei geht das Lernen vom Lernenden selbst aus, so beruht der Ansporn zum Lernen auf Neugierde und Interesse. Auch liegt es am Lernenden, wichtige Informationen auszufiltern und sie nach Prioritäten zu ordnen. Durch das Selektieren lernt er auf neue Probleme adäquat zu reagieren und so neue Lösungen finden zu können. Die Lehrenden sind nunmehr zu Begleitern geworden, die den Lernenden unterstützen.

2.2.1 Umsetzung bei Lernsoftware

Bei Umsetzung dieser Theorie muss besonders darauf geachtet werden, dass die Software tatsächlich beim Lösen von Problemen hilft und so zwischen dem Lernenden und dem Programm ein Dialog entsteht, der auf Fehler aufmerksam macht und weitere Hilfestellungen liefert. Dies erfordert jedoch eine sehr komplexe Softwarestruktur.

¹ Skinner 1958.

² Holzinger 2001, Bd. 2, S. 137.

³ http://de.wikipedia.org/wiki/Entdeckendes_Lernen

2.2.2 Kritik

Ein großer Nachteil scheint zu sein, dass schwächere Lerner bei der Anwendung des „*entdeckenden Lernens*“ leicht überfordert sind, da sie mit ihrer Eigenständigkeit nicht umgehen können oder aber nicht in der Lage sind, Prioritäten zu setzen. Auch geht der Kognitivismus nicht auf die jeweilige psychische Verfassung des Lernenden ein und stützt sich, wie auch der Behaviorismus auf „*Repräsentationen mit logischen Beziehungen und Wahrheitsbedingungen, die sich aber wieder auf eine objektive Wirklichkeit stützt*“⁴.

2.3 Konstruktivismus

Konstruktivisten sind der Meinung, dass es keine objektive Wahrnehmung der Umwelt gibt, da diese nur ein Gebilde aus Informationen ist, die im Gehirn gesammelt werden. Somit gibt es nur eine subjektive Wahrnehmung. Mit dieser Grundsäule begeben sie sich in direkten Widerspruch zu den Kognitivisten.

„Lernen ist aktive Wissenskonstruktion in Verbindung mit bereits bestehendem Vorwissen.

Lernen ist individuell – der jeweilige Lernweg ist nicht vorhersehbar.“⁵

Im Konstruktivismus stehen die internen Verstehensprozesse im Mittelpunkt. Der Lernende eignet sich das Wissen mit einer selbstgewählten Lernmethode an. Es wird davon ausgegangen, dass er das Wissen leichter behält, da er es sich freiwillig aneignet, indem er wahrnimmt, interpretiert und konstruiert. Der Lehrende hat lediglich die Aufgabe ihn zu unterstützen.

Auch widersprechen die Konstruktivisten den Behavioristen dahingehend, dass sie der Meinung sind, es gebe keine Verbindung zwischen „*Stimulus und Response*“⁶. Im Konstruktivismus wird dies auf „*interne Verstehensprozesse*“ zurückgeführt und Vorwissen eine große Bedeutung beigemessen, da man davon ausgeht, dass neues Wissen nur auf bereits vorhandenem aufbauen kann. Auch sollte Wissen nur mit realem Hintergrund vermittelt werden.

2.3.1 Umsetzung bei Lernsoftware

Im Gegensatz zu anderen Lerntheorien sind die Ideen des Konstruktivismus nicht so einfach mit den Methoden der modernen IT umzusetzen. Sie können nur zur Anregung oder Motivation genutzt werden, nicht jedoch zum Lernen selbst, weil das

⁴ Holzinger 2001, Bd. 2, S. 145.

⁵ Holzinger 2001, Bd. 2, S. 149.

⁶ Siehe: 2.1 Behaviorismus, S. 2.

Lernen im Verständnis der Konstruktivisten vom Individuum selbst gesteuert wird und die Umwelt keinen Einfluss darauf hat. Durch eine individuelle und somit nicht vorhersagbare Vorgehensweise des Lernenden lassen sich schwer Programme und Lernstrategien entwickeln, die flexibel genug sind.

Jedoch muss man Lernsysteme nicht ausschließlich als Mittel zum Lernen verstehen, sondern kann sie auch als „*Informations- und Werkzeugangebot*“⁷ sehen. Da sich der Lernende die Informationsquellen selbst aussucht, bilden das Internet und die neuen Medien ein weites Feld für vielfältige Recherchen und Möglichkeiten, Aufgaben auf unterschiedliche Weise zu lösen. Für Lehrende, die den Lernenden mehr Verantwortung und Kontrolle über den Lernprozess geben wollen, bietet das Internet besonders viele Anreize. Versuche zeigen, dass sich Schüler durch das Arbeiten mit Informationen, wie zum Beispiel bei der Erstellung einer Website zu einem bestimmten Thema, die entsprechenden Informationen aneignen. Clark⁸ kam 1983 zur Erkenntnis, dass Hochbegabte am ehesten einen strukturierten und geleiteten Lernprozess wählen, obwohl sie mit den *entdeckenden* Methoden am besten zurecht kommen würden. Auf der anderen Seite entscheiden sich weniger Begabte meistens für Lernarten, bei denen entdeckendes Lernen beinhaltet ist, obwohl sie am meisten von direktiven Methoden profitieren würden. Diese Aussage wurde aber bereits wenig später in Verbindung mit Hypertext in Frage gestellt.

2.3.2 Kritik

Im Konstruktivismus ist der Lehrende kaum verantwortlich für den Lernfortschritt des Lernenden, weil dieser allein sein Lernen beeinflusst, was eine gewisse Reife voraussetzt.

⁷ Holzinger 2001, Bd. 2, S. 163.

⁸ Holzinger 2001, Bd. 2, S. 190.

3 Überblick über Softwaretypen

Im folgenden Abschnitt soll versucht werden, die große Zahl unterschiedlicher Applikationen, die zu pädagogischen Zwecken verwendet werden, einigen Basistypen zuzuordnen und so eine einfache Typologie, von in unterschiedlichen Bereichen der Pädagogik zu Anwendung kommender Software, auszuarbeiten. Die Liste ist zweifellos erweiterbar, dies rührt vor allem daher, dass es in diesem Bereich viele hybride Lösungsansätze gibt, die sich aus mehreren Softwarelösungen zusammensetzen.

3.1 *Präsentations- und Visualisierungssoftware*

Ein Typ von Programmen, mit denen man inzwischen in vielen Bereichen zu tun hat, sind gängige Präsentations- und Visualisierungsprogramme. Sie kommen zum Einsatz, wenn herkömmliche Präsentations- und Darstellungswerkzeuge an ihre Grenzen stoßen, wie zum Beispiel bei statischen Medien. Besonders bei Vorträgen und Präsentationen wird diese Art von Software zur Darstellung von Fakten benutzt, aber auch um real nicht Vorhandenes den Schülern in Form von Modellen und Animationen näher zu bringen.

3.2 *Drill-and-Practice-Programme*

Hierbei handelt es sich um die einfachste Art instruktioneller Software, wie sie seit der Frühzeit des computerbasierten Lernens existiert. Ein Beispiel für eine typisch behavioristische Software ist ein Vokabeltrainer. Der User erhält sofort nach der Eingabe der Antwort ein Feedback über deren Richtigkeit. Bei falschen Antworten benötigt man zumeist eine gewisse Anzahl von richtigen, um wieder auf dem selben Stand zu sein und für richtige Antworten werden zum Beispiel grafische Animationen, Tonausgaben oder Ähnliches dem Lernenden als Belohnung präsentiert. Es darf dabei der Überraschungseffekt nicht vernachlässigt werden, da dies ein starker Ansporn ist.

Beim *practice* Teil des Programms werden bereits durchgenommene Aufgaben wiederholt, mit zusätzlichen Erläuterungen zur Festigung.⁹ Diese Art von Software zeichnet sich in aller Regel durch vergleichsweise geringe Komplexität und

⁹ Aus diesem Softwaretypus entwickelte sich später die Courseware, die jedoch das behavioristische Konzept aufgegeben hat. Schulmeister 2002, S. 67.

Flexibilität aus. Sie ist immer nur für ganz konkrete Bereiche einsetzbar. Die Zahl an Beispielen ist sehr groß.

3.3 Tutorielle Systeme

Einen Schritt näher in Richtung komplexere und umfassendere Lösungen stellen die tutoriellen Systeme dar. Hier tritt der Computer als „Lehrer“ auf, jedoch vermittelt er *„kein Faktenwissen, sondern prozedurales Wissen (z.B. Regeln)“*¹⁰. Der Lernerfolg kann durch Fragen überprüft werden. Dieser Typ von Softwarelösung kombiniert sowohl Merkmale der Präsentationssoftware als auch der Drill-and-Practice-Software.

Solche Lernsysteme basieren auf so genannten Tutorials, die wie folgt unterteilt werden:

- Schrittweise Einführung in Programme, bei der sich der User passiv verhält.
- Lineare Lernprogramme. Hier erfolgt nach der Wissensvermittlung eine Abfrage.
- Multifunktionale Lernprogramme, bei denen der Benutzer die Reihenfolge und die Art der Abfrage bestimmt.

Die tutoriellen Systeme stellen, nicht nur an Schulen und im universitären Bereich, sondern auch in der Wirtschaft den in den letzten Jahren wohl am häufigsten benutzten Typus dar, da er relativ einfach und kostengünstig zu erstellen ist.

3.4 Intelligente tutorielle Systeme (ITS)

Eine Weiterentwicklung des vorangegangenen Typs stellen so genannte intelligente tutorielle Systeme dar. Als intelligent werden sie bezeichnet, da die Programme Benutzerprofile von ihren „Schülern“ anlegen und diese daraufhin spezifisch fordern und fördern können. Diese Systeme bauen bereits bis zu einem gewissen Grad auf Erkenntnissen der AI¹¹-Forschung auf, sind schwer herzustellen und entsprechend kostspielig. Sie sind daher eher selten anzutreffen und wenn, dann nur in der Forschung. Da das Programm selbstständig Lernpläne ausarbeitet und Übungen vorschlägt, wird dem Lernenden erneut eine passive Rolle zugeteilt.

¹⁰ Holzinger 2001, Bd. 2, S. 227.

¹¹ Artificial Intelligence; auf deutsch KI=künstliche Intelligenz.

Die meisten intelligenten tutoriellen Systeme bestehen aus:

Wissensmodell¹²

Es enthält das Wissen zu einem gewissen Themengebiet. Es ist in Form von Listen, Wissensstrukturdiagrammen oder Regeln geordnet. Die einzelnen Attribute sind durch Relationen miteinander verbunden. Das System ist in der Lage, Aufgaben zu lösen und zu erklären und die Lösungsvorschläge des Lernenden zu analysieren.

Lernermodell¹³

Seine Aufgabe ist es, den aktuellen Wissensstand des Lernenden zu definieren und zu analysieren. Dieser kann entweder als Teilmenge des Wissensmodells oder als Abweichung von Expertenwissen dargestellt werden.

Tutorenmodell¹⁴

Die Aufgabe dieser Komponente ist die Simulation des Entscheidungsverhaltens eines Lehrers, des Prozesses seiner Entscheidungen und seine Instruktionen. Es hat die Aufgabe zu bestimmen, welches Lernmaterial wann und wie präsentiert wird. Dieser Teil des Programms ist sozusagen der Pädagoge.

Kommunikationskomponente¹⁵

Sie ist verantwortlich für die Kommunikation zwischen Lernenden und Lehrenden. Es gibt verschiedene Arten der Interaktion:

- Sokratischer Dialog: Das System stellt dem Lernenden Fragen und ermittelt gemäß seiner Antworten Fehler.
- Coaching: Das System greift erst ein, wenn der Lernende darum bittet.
- Learning by doing: Das System selbst ist aktiv. Der Lernende ist aufgefordert, Informationen auszuwählen. Die Abweichungen vom Wissensmodell werden registriert.
- Learning while doing: Das System gibt nur von Zeit zu Zeit hilfreiche Hinweise.

3.5 Simulationsprogramme

Simulationsprogramme sind Applikationen, die virtuelle Welten erzeugen, in denen Lernende zu pädagogischen Zwecken agieren sollen. Solche Programme simulieren konkrete Anwendungs- und Handlungssituationen. Zuvor erworbenes Wissen soll in möglichst realitätsnahen Situationen angewendet werden. Der Erfolg der Programme hängt besonders von der qualitativen Umsetzung ab und stellt sehr oft auch

¹² auch Expertenmodell.

¹³ auch Diagnosemodell.

¹⁴ auch pädagogisches Modell.

¹⁵ auch Interface.

besondere Ansprüche an die eingesetzte Hardware. Diese Art von Programmen kommt auch in der Unterhaltungsindustrie stark zur Geltung. Beispiele sind Flugsimulatoren wie *Microsoft Flight Simulator* oder *Sim City 2000*, welches den User Städte errichten lässt und ihn beauftragt, diese zu managen.

Grundsätzlich werden hier zwischen Simulationen mit geringerer Komplexität, um Wechselwirkungen besser analysieren zu können, und hoher Komplexität, um Einblicke in Situationen ohne Vorhersagbarkeit gewinnen zu können, unterschieden. Derartige Applikationen wären auch in vielen Bereichen der Pädagogik wünschenswert, die Umsetzung scheitert jedoch nach wie vor nicht nur an den finanziellen, sondern vor allem an den technischen Grenzen der Softwareersteller.

3.6 Mikrowelten und Modellerstellung

Bei dieser Software handelt es sich um eine Weiterentwicklung von Simulationen. Mikrowelten integrieren jedoch zusätzlich noch konstruktivistische Ansätze, da der Lernende nicht auf die vorgegebenen Aufgaben angewiesen ist, sondern sich selbst neue Lernziele durch die Modellierung einer neuen Situation oder eines Problems stecken kann. Dies setzt viel Eigeninitiative und Selbstverantwortung voraus. Ein schönes Beispiel hierfür stellt die Software des Projekts GEONET der Uni Bayreuth dar, welche den Studierenden einen experimentellen Zugang zur dynamischen Geometrie eröffnet.

3.7 Hypermedia-Programme

Diese Programme bieten dem User an, sich völlig frei durch die Lerninhalte zu navigieren, wie beim Surfen im Internet. Als Beispiel ließe sich die *Encyclopaedia Britannica* nennen, die neben einer großen Anzahl von Einträgen auch schon Zugriff ins Internet bietet. Diese Art von Programmen kann aber zum „Lost in Hyperspace“¹⁶-Problem führen.

3.8 Lernspiele

Lernspiele können zumeist zu den Simulationenprogrammen gezählt werden. Man unterscheidet zwischen Software mit fest definierter¹⁷ und einer nicht eindeutig

¹⁶ Der Lernende verliert das gewünschte Lernziel aufgrund des reichhaltigen und unüberschaubaren Informationsgehalts aus den Augen.

¹⁷ Game (deutsch: Spiel).

definierten und fehlenden Gewinnsituation¹⁸. Beide Spieltypen sind für die Persönlichkeitsbildung in der Kindheit sehr wichtig.¹⁹ Es werden Rollen eingenommen und verschiedene Situationen nachgespielt.

Gute Lernsoftware versucht Spielhandlungen mit Lernkomponenten zu verbinden, um so gleichzeitig Wissen als auch Spielspaß zu vermitteln.

¹⁸ Play (deutsch: Spielzeug).

¹⁹ Diese Aussage bezieht sich generell auf die Spiele von Kindern.

4 Aspekte der Softwareerstellung

„Im Mittelpunkt eines Softwareprojektes steht die Lösung eines didaktischen Problems. Multimedia muss als (ein mögliches) Element der Problemlösung betrachtet werden.“²⁰

Die Erstellung von Software für einen bestimmten Bereich stellt einen äußerst komplexen Vorgang dar. Zusätzlich kompliziert werden Softwareprojekte im pädagogischen Bereich dadurch, dass sie an der Schnittstelle einer Reihe unterschiedlicher wissenschaftlicher Disziplinen angesiedelt sind. In der Regel braucht man zur erfolgreichen Umsetzung eines solchen Projekts einer Reihe von Spezialisten: neben den Pädagogen müssen Fachleute aus den Gebieten Informatik, Design und Management miteingebunden werden. Sollen multimediale Inhalte integriert werden, wird das vollständige Inventar an Audio- und Video-Know-how gebraucht, das üblicherweise beim Produzieren von Ton- oder Filmmaterialien vonnöten ist.

4.1 Entwicklungsprozess

Von entscheidender Bedeutung ist die exakte Definition der Projektvorgaben sowie das Festlegen von Zuständigkeiten und Abläufen. Projektvorgaben werden dabei üblicherweise in Form von Pflichtenheften getroffen. Je genauer zu Beginn eines Projekts festgelegt wird, was die Software können soll, desto einfacher gestaltet sich die Projektabwicklung.

Im Laufe der Zeit entstanden eine Vielzahl von Prozessablaufmodellen, die sich oft kaum voneinander unterscheiden. In den 70er Jahren entstand das klassische Wasserfallmodell, das zum Ziel hatte, den Entwicklungsprozess planbar zu machen *„innerhalb von festen Budget- und Zeitrahmen mit kontrollierter Qualität“²¹* und mit einer festgelegten Arbeitsteilung.

Der gesamte Entwicklungsprozess wird in vier Phasen unterteilt:

- Analyse und Definition
- Entwurf (Design und Spezifikation)
- Implementierung
- Test

Ein großer Nachteil dieses Modells ist, dass man um die Qualität der einzelnen Phasen zu gewährleisten, erst dann zur Nächsten fortschreiten kann, wenn die letzte

²⁰ Holzinger 2001, Bd. 2, S. 233.

²¹ Holzinger 2001, Bd. 3, S. 28.

Phase erfolgreich abgeschlossen ist. So können Veränderungsvorschläge während des Projektverlaufs nicht mehr einbezogen werden und auch Fehler, die in der Anfangsphase nicht bemerkt wurden, erst spät entdeckt werden. Einerseits kann der Fehler so tief liegen, dass das ganze Konzept neu überarbeitet werden muss, andererseits kann sich der ganze Zeitplan verzögern, was zusätzliche Kosten erzeugt und zu Lasten der Testphase geht. Auf dem Wasserfallmodell bauen eine Vielzahl von anderen Modellen auf, die sich zum Beispiel dadurch unterscheiden, dass für das System und die einzelnen Teile ein Prozessverlauf durchgearbeitet wird, wodurch quasi das Programm langsam aufgestockt wird und so auch Fehler in den einzelnen Komponenten sofort entdeckt werden, die sofort behoben werden können und nicht erst am Ende, wenn das Programm beinahe fertig ist.

Aufgrund der vielen Nachteile wurde das Wasserfallmodell auf verschiedenste Weise modifiziert. Eine häufige benutzte Variante gliedert sich in:

Planungsphase

Was wird gemacht?²² Ist die Umsetzung möglich? Welchen Verlauf hat die Entwicklung? Es wird ein Projektplan erstellt, der die Kosten und Fristen enthält.

Definitionsphase²³

Erörterung der Anforderungen an die zu entwickelnde Software. Anlegung eines präzise formulierten Pflichtenhefts²⁴.

Entwurfsphase (Design)

Die Aufgabe des Entwurfs ist eine softwaretechnische Lösung zu erarbeiten. Dies beinhaltet bereits die grobe Planung von Komponenten.

Implementierungsphase

Erst in dieser Phase werden die Komponenten ausprogrammiert, wodurch der Quelltext entsteht. Zur Bewahrung des Überblicks und der Planung sollte auch eine Dokumentation angelegt werden.

Um die Prozessmodellierung zu vereinfachen und um objektorientierte Softwaresysteme zu beschreiben, wurde UML²⁵ entwickelt.

²² Hier sollte ein sogenanntes Lastenheft erstellt werden, welches nicht so detailreich wie das Pflichtenheft sein und nur ein grobes Gerüst enthalten sollte.

²³ Die Aufgabenstellungen aus der Planungsphase dienen als Input.

²⁴ Alles, was nicht im Pflichtenheft erwähnt ist, wird nicht realisiert.

²⁵ Unified Modeling Language.

4.2 Konzipierung

Bei der Erstellung einer Lernsoftware muss man sich, genau wie bei anderen Formen zielgerichteter medialer Kommunikation – wie zum Beispiel in der Werbung – bevor es zur eigentlichen Umsetzung kommt, bewusst werden, welcher Zielgruppe man welche Lerninhalte, mit welchen Lehrzielen, vermitteln will.

4.2.1 Zielgruppe

Besonders wichtig ist es, die Zielgruppe zu bestimmen. Um einerseits das Benutzerfeld nicht allzu sehr einzuschränken, sollte man den Kreis potentieller Konsumenten nicht allzu sehr einengen. Wenn jedoch der Kreis der Abnehmer zu weit gefasst wird, kann dies dazu führen, dass potentielle User sich nicht angesprochen fühlen und so markttechnisch nicht erreicht werden können.

Es macht auch einen großen Unterschied, ob man als Zielgruppe die Angestellten einer bestimmten Firma oder eines Betriebs benutzt oder einen „normalen“ Käufer in einem Softwaregeschäft, der sich auf einem bestimmten Wissensgebiet fortbilden will. Letzterer ist viel schwerer zu erfassen und in eine Zielgruppe einzuordnen. Zur näheren Charakterisierung der Lernenden können folgende Daten hilfreich sein.

4.2.1.1 Soziodemographische Daten

Hierzu kann man neben den Informationen über Alter und Geschlecht auch den sozialen Status und Bildungsstand zählen. Für den Fall, dass man Lernsoftware für einen Betrieb schreibt, ist Wissen über die Positionen und Funktionen der User innerhalb der Firma von Bedeutung. Im Vergleich zum Kunden in Softwaregeschäften, sind Informationen über Betriebsangestellte leichter zu erhalten und der Ersteller der Software-Lernumgebung kann das Programm besser an den User anpassen. Der Bildungsstand lässt womöglich Schlüsse auf das Allgemeinwissen und den Sprachstil zu.

4.2.1.2 Vorwissen und relevante Erfahrungen potentieller User

Die Frage nach dem bereits angeeigneten theoretischen Wissen und den praktischen Erfahrungen ist besonders wichtig, da der Lernerfolg umso größer ist, je besser der Lernstoff an den Lernenden angepasst wird. So kann man versuchen einerseits Langeweile, andererseits Überforderung aufgrund von Mangel an Wissen auszuschließen.

4.2.1.3 Lerngeschichte der Zielgruppe

Hierzu werden die Erfahrungen gezählt, die der User bereits mit Fortbildungsmaßnahmen hat. So stellt sich zum Beispiel die Frage, wie viel Wissen er sich bereits im Umgang mit computer- beziehungsweise webbasierten Lernen angeeignet hat und ob er überhaupt schon einmal am Computer gearbeitet hat. Auch sollten User, die seit längerer Zeit keine Fortbildungen mehr absolviert haben und so außer Übung gekommen sind, anders angesprochen werden als routinierte Lerner.

4.2.1.4 Motivation und Einstellung zum Inhalt

Neben den herkömmlichen Arten die Lernenden zu motivieren, sollten Lernende, die sich nicht selbst entschieden haben mit Medien zu arbeiten, anders angesprochen werden als jene, die keine Angst vor eigenem Versagen bei Programmabläufen haben.

4.2.2 Lernziele und Lerninhalte

Über diese Informationen sollte man sicherlich bereits vor der Planung in groben Zügen Bescheid wissen. Im Zuge der Konzipierung sollte man jedoch klar definieren welche Ziele angestrebt werden und wie ausführlich, weitverzweigt und vor allem wie der Lernstoff aufgebaut ist. Diese Gedanken sollten auch klar festgehalten werden, gut durchdacht sein und der Ersteller sollte diese bei ihrer Arbeit immer vor Augen haben.

Zusätzlich sind noch Lerngewohnheiten und Lerndauer der User, deren bevorzugte Lernorte und Medienzugang von Bedeutung. Auch sind Informationen über die Herkunft der Zielgruppen wichtig, um Rücksicht auf ihre kulturellen Besonderheiten und Sprachenkenntnisse nehmen zu können. Diese Faktoren werden aber wahrscheinlich nur eine Rolle spielen, wenn die Software international eingesetzt werden soll, wie zum Beispiel in großen Konzernen, mit Zweigstellen in anderen Ländern.

4.3 *Usability*²⁶ und Software-Ergonomie

Im Laufe der Entwicklung von Computern und Software lässt sich klar eine Tendenz hin zu größerer Nähe zum natürlichen Verhalten des Menschen feststellen. Die Maschine muss dem Menschen entgegenkommen, nicht umgekehrt. Die

²⁶ Engl.: Benutzerfreundlichkeit, Benutzbarkeit.

verschiedenen Aspekte, die im Zusammenhang mit der Benutzerfreundlichkeit von Software stehen, haben bereits in den 90er Jahren des vorigen Jahrhunderts viel theoretisches Interesse ausgelöst. Während Programmierer sich mit dem System auseinandersetzen, müssen Designer die Benutzer und deren Situation verstehen. Das User-Interface bildet die zentrale Schnittstelle bei der Arbeit von Mensch und Maschine.

Kriterien für die Benutzerfreundlichkeit:

Transparenz

Diese beinhaltet, dass der User einen größtmöglichen Überblick über den jeweils aktuellen Zustand des Programms hat, wie zum Beispiel ob es noch an einer Verifikation einer Antwort arbeitet.

Konsistenz

Das Programm sollte auf gleichartige Eingaben ähnlich reagieren und natürlich sollte das Design gleichbleibend sein, da von Seiten des Users auch eine Gewöhnung eintritt.

Toleranz

Fehler sollten behebbar sein, ohne einen Prozess neu zu starten. Ebenso sollten Eingaben durch Fehler nicht verloren gehen. Bestenfalls sollte das Programm auf Fehler verschieden reagieren und diese wenn möglich sogar abfangen.

Kompatibilität

Die Software sollte an seinen Benutzer angepasst sein, so dass zum Beispiel das richtige Fachvokabular benutzt wird. Auch sollte es eine Übereinstimmung zwischen der Darstellungsform auf dem Bildschirm und der auf gedruckter Vorlage geben.

Unterstützung

Software sollte eine Hilfefunktion haben, die sowohl einen Index als auch eine Rückfragemöglichkeit enthält. Die meisten Programme verfügen bereits als Standardausrüstung über solche Hilfskomponenten.

Flexibilität

Die Software sollte flexibel an den User anpassbar sein. So sollte dieser zum Beispiel in der Lage sein, Bildschirmfarben und Schriftgrößen verändern zu können. Und auch die Abfolge der Arbeitsschritte und die Ablaufgeschwindigkeit.

Partizipation

Der User sollte in der Lage sein, über Feedback und Korrekturvorschläge an neueren Versionen mitarbeiten zu können.

5 Die Geschichte des multimedialen Lernens und die Zukunft des E-Learnings

Eine wohlbekannte Tatsache ist, dass es Lernmaschinen bereits vor der Erfindung von Digitalrechnern gab. Mit jeder Entwicklung eines neuen Mediums verbanden sich große Erwartungen am Lernsektor. Nach der Erfindung des Radios wurde das Rundfunklernen propagiert, bei der des Fernsehers, das Telelernen.²⁷ Nach der Etablierung des PCs wurde das Computer-Based-Training forciert. Nun mit der fortschreitenden Entwicklung des Internets und der immer größer werdenden technischen Möglichkeiten werden das Web-Based-Training und die Bildung von virtuellen Universitäten zu immer häufiger benutzten Schlagworten. In den folgenden Kapiteln wird die Geschichte des Lernens mit Medien beleuchtet. Besonderes Augenmerk soll dabei auf die Entwicklung vom Computer-Based-Training zum Web-Based-Training gelegt werden.

5.1 Die Zeit vor der Erfindung des PCs

Die ersten Lernmaschinen waren alle mechanischer Natur. So waren auch die Möglichkeiten im Vergleich zu heutigen Computern stark begrenzt und führten auch damals nur zu sehr begrenzten Erfolgen.

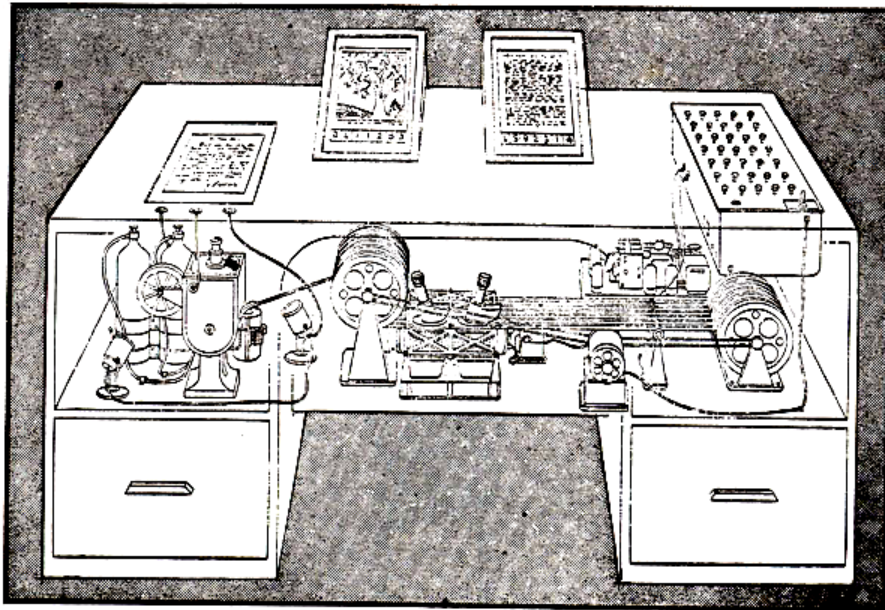
Einer der frühen Pioniere der Informationstechnik in der ersten Hälfte des 20sten Jahrhunderts war Vannevar Bush²⁸ mit seinem Projekt *Memex*²⁹, das er 1945 vorstellte, welches aber auf Grund von fehlender Technologie nicht umgesetzt werden konnte. Bush ging davon aus, dass das menschliche Gehirn anhand von Assoziationen arbeitet. Sein *Memex* sollte den ganzen Arbeitstag filmen und die relevanten Informationen archivieren. Die Maschine sollte als Unterstützung und Erweiterung für das Gedächtnis und das Assoziationsvermögen dienen. Dies konnte durch moderne Technologien wie das Internet realisiert werden.³⁰

²⁷ Holzinger 2001, Bd. 2, S. 175.

²⁸ Vannevar Bush (*1890; m. 1974). Siehe auch: http://de.wikipedia.org/wiki/Vannevar_Bush

²⁹ Abkürzung für „memory expander“. Holzinger 2001, Bd. 2, S. 179.

³⁰ Siehe 5.3. Von fest-verdrahteten Lösungen zu mehr Flexibilität: Hypertext, S. 18.



Vannevar Bushs Memex
<http://kelty.rice.edu/375/lectures/bush>

In den 1950er Jahren entwickelte Skinner³¹ Lehrmaschinen, deren theoretischen Ansätze bis zur Entwicklung von Lernsoftware in Vergessenheit geriet. Er unterschied zwischen linearen Lernprogrammen und verzweigten. Bei ersterer wurde Lehrmaterial so positioniert, dass die Chance des Lernenden die richtige Antwort zu nennen größer war. Als Belohnung für eine richtige Antwort, darf man sich mit der nächsten Lerneinheit befassen.³² Bei zweiterer wird bei einer falschen Antwort in zwischenliegende oder ergänzende Einheiten „weitergeleitet“.

5.2 Die ersten Computer

Durch die raschen Fortschritte, die die Sowjet Union in den 60er Jahren in der Raumfahrt machte, begannen die USA³³ viel Geld in Forschung und Bildung zu investieren, in digitale Trainingsprojekte, so etwa in die Projekte TICCIT und PLATO.

5.2.1 TICCIT³⁴

Die Lernenden lernen an einem Terminal, das aus einem Farbmonitor, Lautsprechern, einer speziellen Tastatur, Lichtgriffel und einem Abspielgerät für Videobänder besteht. Der Lernstoff wird in drei Schritten präsentiert: als Regel, als

³¹ Burrhus Frederic Skinner (*1904; m. 1990). Siehe: www.bfskinner.org

³² http://en.wikipedia.org/wiki/B._F._Skinner

³³ National Science Foundation.

³⁴ Time-Shared Interactive Computer Controlled Information Television. Holzinger benutzt die Abkürzung TCCIT, wohingegen man im Internet nur auf TICCIT stößt.

Beispiel und als Übung, diese werden durch Feedback ergänzt. Der Lernende kann selbst den Schwierigkeitsgrad wählen und auch jederzeit Hilfe anfordern.³⁵

5.2.2 PLATO³⁶

An der Entwicklung von *PLATO* wurde bereits ab den 60er Jahren gearbeitet und erst später wurde es wie das *TICCIT* durch den NSF³⁷ subventioniert. Im Vergleich zu diesem ist es jedem Lehrer durch die Autorensprache *TUTOR* möglich selbst Anwendungen zu entwickeln. Die Oberfläche auf der der Lernende arbeitet, besteht aus einem berührungssensitiven Plasmabildschirm und einer Tastatur. Über dieses Terminal konnten auch noch andere Geräte, wie Projektoren oder Laborgeräte bedient werden. Verbunden waren diese Oberflächen durch einen Steuerrechner, der bis zu 1000 weitere versorgen konnte. Dies bildet eine „vollwertige, vernetzte, multimediale Lernumgebung“³⁸ die noch heute aktiv ist und an deren Entstehungsort³⁹ noch heute 400 Terminals bereits stehen. Insgesamt sind 1300 Terminals miteinander verbunden.

PLATO dient aber nicht nur als erste größere Plattform für multimediales Lernen, sondern auch als erste „Online Community“ in der die ersten Online Spiele ihren Ursprung nahmen. 2004 wurde eine neue Online Version von *PLATO* veröffentlicht, die sehr schnell ihre früheren Benutzer zusammenführte.⁴⁰

5.3 Von fest-verdrahteten Lösungen zu mehr Flexibilität: Hypertext

Bald fehlte den Programmen, wie auch schon vielen Lernmaschinen vor ihnen, die Benutzerfreundlichkeit. Zur Erstellung von einer Stunde *Courseware* musste der Lehrende 100 Arbeitsstunden in Kauf nehmen und es war ein Kostenaufwand von 20.000 US-Dollar notwendig.⁴¹ Die schlechte Benutzbarkeit resultierte gewöhnlich aus den mangelnden technischen Fortschritten auf dem Gebiet der Hardware.

Durch Hypertext wurde erstmals nicht lineares Lesen ermöglicht.

³⁵ Schulmeister 2002, S. 100.

³⁶ Programmed Logic for Automatic Teaching Operations.

³⁷ Siehe: FN 33.

³⁸ Holzinger 2001, Bd. 2, S. 182.

³⁹ University of Illinois.

⁴⁰ <http://en.wikipedia.org/wiki/PLATO>

⁴¹ Holzinger 2001, Bd. 2, S. 183.

5.3.1 NLS/Augment

Durch die fortschreitende Entwicklung von Computern und die immer größer werdenden Möglichkeiten, frühere Ideen in die Realität umzusetzen, entwickelte ab 1963 Douglas Engelbart⁴² das System *NLS/Augment* am Stanford Research Institute. Es wurden Dateien hierarchisch angeordnet und durch Links miteinander verknüpft. Bei Augment handelt es sich um das älteste Hypertext- System. *Augment* verfügte bereits über eine „*Maus als Eingabemedium, eine Mehrfenstertechnik mit ‚parallel processing‘, die Möglichkeit der Verknüpfung heterogener Materialien über Zeiger und integrierter Einsatz von Grafik*“⁴³

5.3.2 XANADU

Im Jahr 1965 startete Ted Nelson⁴⁴ das Projekt *XANADU* an dem bis heute gearbeitet wird. Das angestrebte Ziel war es alle Publikationen dieser Welt zu erfassen und untereinander zu vernetzen. Bei *XANADU* handelte es sich erstmals um ein „*echtes Hypertextsystem mit point- and- click*“⁴⁵. Der Begriff Hypertext wurde durch dieses Projekt erstmals eingeführt.

Nelson schwebte bereits ein *Client-Server-System* vor, welches eine Trennung von User und Server bedeutete. Bis zu diesem Zeitpunkt waren nur *Master-Slave-Systeme* im Umlauf gewesen, bei denen sich ein ganzes System auf einen Zentralrechner stützte, der mit Endgeräten verbunden war. Die Verarbeitung fand jedoch nur im Zentralrechner statt.

5.3.3 Hypertext Editing System (HES)

Laut Andreas Holzinger handelte es sich bei *HES* um das erste komplett funktionsfähige Hypertextsystem, das über einen Browser, eine Funktionstastatur und einen Lichtgriffel, eine Hilfefunktion und In-text-Links verfügte. 1968 erschien das Nachfolgesystem *FRESS*⁴⁶. Man unterschied bei diesem Programm zwischen:

- Tag: eine einseitige Verbindung auf ein einzelnes Element, wie zum Beispiel eine Anmerkung, die in einem eigenen Fenster geöffnet wurde.
- Jumps: Verbindungen in beide Richtungen, die hin und wieder retourführten.

⁴² Douglas Engelbart (*1925) wurde von Ideen von Vannevar Bush (siehe 5.1 Die Zeit vor der Erfindung des PCs S. 16) beeinflusst. Schulmeister 2002, S. 225.

⁴³ Holzinger 2001, Bd. 2, S. 185.

⁴⁴ Ted (Theodor Holm) Nelson (*1939); Siehe <http://xanadu.com.au/ted>

⁴⁵ Holzinger 2001, Bd. 2, S. 187.

⁴⁶ File Retrieve and Editing System.

5.4 Lernsoftware unter dem Einfluss des Kognitivismus

Unter dem Einfluss des Kognitivismus entstanden zahlreiche Programme, die sich dadurch auszeichneten, dass der Lernende die Stoffgebiete versteht und er selbst nicht als passiver User gesehen wird.

5.4.1 Adaptive generative Systeme

Diese Art von Programmen ist in der Lage, Aufgaben unterschiedlicher Schwierigkeitsgrade zu generieren und daraufhin die Lösung zu beurteilen. Die Idee dieser Systeme geht davon aus, dass die angepassten Schwierigkeitsstufen, dem Lernenden ein effektiveres Lernen ermöglichen.

MAIS⁴⁷

Bei diesem Programm handelt es sich um eine Kombination von computerunterstützter Instruktion und einem ITS⁴⁸. Der Computer beobachtet, den Lernenden und kalkuliert über einen Algorithmus⁴⁹, ob der Lernende die Aufgabe bestehen wird. Nach jeder Antwort wird erneut die Kompetenz des Lernenden berechnet.

5.4.2 Intelligente tutorielle Systeme⁵⁰

Das Programm *SCHOLAR*⁵¹ basiert konzeptionell auf dem sogenannten sokratischen Dialog. Es war bereits so angelegt, dass der Teil des Programms, der die Antworten überprüfte und daraus Folgerungen zog, unabhängig vom aktuellen Wissensgebiet funktionierte.

SOPHIE⁵²

Dieses Programm ist ein Modell für elektronische Schaltkreise. Die Lernenden versuchen Aufgaben zu lösen und können gegebenenfalls Fragen an das System stellen, das diese daraufhin beantwortet. Dies bedeutet, dass die tutorielle Komponente des Programms erst nach Aufforderung eingreift. Weiters gab es auch bei *BUGGY*⁵³, ein Programm zur Lösung mathematischer Probleme, noch kein so genanntes Coaching. Bis *SOPHIE III* wurde das Programm durch einen Coach,

⁴⁷ Minnesota Adaptive Instructional System.

⁴⁸ Siehe 3.4. Intelligente tutorielle Systeme (ITS), S. 7.

⁴⁹ Beruht auf dem Bayes'schen Netz, das der Repräsentation von unsicherem Wissen und daraus möglichen Schlussfolgerungen dient. Siehe http://de.wikipedia.org/wiki/Bayessches_Netz

⁵⁰ Siehe 3.4. Intelligente tutorielle Systeme (ITS), S. 7.

⁵¹ 1970 von Carbonell entwickelt. Thema war die Geographie Südamerikas. Siehe Martens 1970.

⁵² 1974 von Brown und Burton entwickelt. Siehe <http://mm-werkstatt.informatik.uni-augsburg.de>

⁵³ 1978 von Brown und Burton entwickelt. Siehe <http://mm-werkstatt.informatik.uni-augsburg.de>

einen Experten als Problemlöser, der auch fähig ist, seine Strategien zu erklären sowie durch ein sprachliches Interface ergänzt. *SOPHIE* hebt sich von anderen *ITS* Programmen dadurch ab, dass es dem Lernenden die Möglichkeit zur Reflexion und Selbstevaluation offen lässt. Auch bietet das Programm eine Komponente an, mit deren Hilfe eigene Experimente durchgeführt werden können.

5.4.3 Dynabook

Ein Vorgänger der graphischen Nutzeroberflächen wurde bereits in den 60er Jahren erdacht und 1972 konnte ein erster Prototyp realisiert werden. Alan Kay⁵⁴, der Hauptverfechter von Dynabook, vertrat als einer der Ersten die Ansicht, dass sich der Computer den Bedürfnissen des Menschen anzupassen habe und nicht umgekehrt. Kay lernte 1968 Seymour Papert⁵⁵ kennen, der versuchte Computer in die Erziehung von Kindern einfließen zu lassen und der mit Logo eine Programmiersprache entwickelt hatte, die primär zu pädagogischen Zwecken geschaffen wurde.



Prototyp des Dynabooks

http://www.artmuseum.net/w2vr/archives/Kay/01_Dynabook.html

5.5 Lernsoftware unter dem Einfluss des Konstruktivismus

5.5.1 Mikrowelten

„Mikrowelten sind geschlossene artifizielle Umgebungen mit eigenen Regeln.“⁵⁶

Diese künstlichen Welten lehnen sich an konstruktivistische Lerntheorien an. So soll in einer vielschichtigen Lernumgebung bereits bekanntes Wissen reflektiert und

⁵⁴ Alan Kay (*1940). Siehe <http://ei.cs.vt.edu/~history/GASCH.KAY.HTML>

⁵⁵ Seymour Papert (*1928). Siehe <http://www.papert.org> oder http://en.wikipedia.org/wiki/Seymour_Papert

⁵⁶ Schulmeister 2002, S. 50.

neues Wissen konstruiert werden. Die Programmierer verpacken das zu erlernende Wissen in Mikrowelten und die Lernenden müssen es entdecken und finden. Aber auch einige Konstruktivisten vertraten die Meinung, dass Mikrowelten als Lernumgebung zu künstlich seien. Die erste Lernumgebung dieser Art war Logo:

LOGO

Dieses von Seymour Papert am MIT⁵⁷ 1967 entwickelte Programmiersystem, hatte die Aufgabe Kindern einen Einstieg ins Programmieren zu erleichtern. Einige grundlegende Prozeduren sind bereits in Logo enthalten, der Benutzer kann jedoch auch neue definieren. Zusätzlich unterstützte Logo eine graphische Einheit, die in der Lage ist Mikrowelten zu erzeugen. Mit Logo konnte der User zum Beispiel Figuren oder bewegliche Modelle kreieren.

Heute kann man auch virtuelle Museen und Landschaften zu Mikrowelten zählen. Oft wird jedoch jegliche Form einer künstlich erzeugten Welt als Mikrowelt bezeichnet, wodurch der Begriff seinen speziellen Sinn verliert und die Eigenschaften, Geschlossenheit und Simulation verliert.⁵⁸ Mikrowelten bieten, nicht nur die Möglichkeit Aufgaben in einer der Realität nachgebildeten Umgebung zu lösen, sondern sich selbst Aufgaben zu stellen.

5.5.2 Erste Hypertextsysteme

Das 1984 von Xerox Parc⁵⁹ entwickelte Hypertextsystem, *NoteCards* zählt zu den sogenannten Mehrfenster- Hypertextsystemen. Wie im Namen des Programms schon enthalten, ist jeder Knoten eine Datenkarte auf die ein Link verweist. Dieser kann jedoch an beliebigen Stellen eingebettet sein. *NoteCard* sollte die Erstellung analytischer Berichte unterstützen.

1987 stellten sich die ersten Erfolge⁶⁰ mit dem Hypertextsystem *Hypercard* von Bill Atkinson⁶¹ ein, das er für die Firma Apple entwickelt hatte und die ersten 5 Jahre den Usern kostenlos zur Verfügung stand. Dieses Programm ist ein Autorenwerkzeug, womit man mit Hypertalk⁶², leicht eigene Hypertextsysteme erstellen kann. Es basiert auf einem Modell, das die Daten in Karten organisierte. Die *files* werden *stack*

⁵⁷ Massachusetts Institute of Technologie.

⁵⁸ Schulmeister 2002, S. 51.

⁵⁹ Siehe <http://www.parc.xerox.com>

⁶⁰ Schulmeister 2002, S. 229.

⁶¹ Bill Atkinson. Siehe http://en.wikipedia.org/wiki/Bill_Atkinson

⁶² „eine leicht zu erlernende Sprache“ (Holzinger 2001, Bd. 2, S. 201).

genannt und jeder *stack* enthält *cards*, die in *layern*⁶³ angeordnet sind, die wiederum übereinander gelegt werden. In den *Cards* ist die Information enthalten und jede *card* kann *buttons*⁶⁴, *fields*⁶⁵ und *paint*⁶⁶ enthalten.

5.6 World Wide Web

Ein wichtiger Schritt in der digitalen Entwicklung stellte die Verbindung einzelner Geräte dar, wodurch die Entstehung kleiner und größerer Netzwerke möglich wurde. Der Aspekt der Kommunikation trat dadurch in den Vordergrund der Entwicklung.

5.6.1 Die Anfänge des Internets

“Der Buchdruck ist die Erfindung des zweiten Jahrtausends; möglicherweise ist das Internet die Erfindung des dritten Jahrtausends.”⁶⁷

Das Internet ging aus dem ARPA⁶⁸-Netz hervor, das vom US-Verteidigungsministerium finanziert worden war. Anfänglich wurde es nur für Experimente und theoretische Untersuchungen benutzt. In dieser Zeit wurden Basisprotokolle für E-Mail (POP) und *file sharing* mittels FTP entwickelt. Ab 1975 musste sich das Projekt finanziell selbst tragen: Nachdem die ersten User Universitäten und Forschungseinrichtungen waren, kamen später auch Computerfirmen hinzu. Während 1984 1000 Rechner am ARPA-Netzwerk hingen, so sind es 3 Jahre später 27.000.⁶⁹ 1989 erhielt das Internet einen rasanten Aufschwung durch die Veröffentlichung des ersten grafikfähigen Webbrowsers Mosaic.⁷⁰

5.6.2 WWW

1989 begann Tim Berners-Lee⁷¹ am Europäischen Kernforschungslabor (CERN⁷²) ein Projekt umzusetzen, das mit Hilfe von Hypertext einen weltweiten Datenaustausch zwischen Wissenschaftlern ermöglichen sollte. Mit der Veröffentlichung der ersten Webseite, schuf er das World Wide Web. Die wohl wichtigste Handlung Berners-Lee neben der Entwicklung des WWW, war wohl auf

⁶³ Ebenen.

⁶⁴ Bedienfelder.

⁶⁵ Text.

⁶⁶ Grafiken.

⁶⁷ Holzinger 2001, Bd. 2, S. 177.

⁶⁸ Advanced Research Projects Agency.

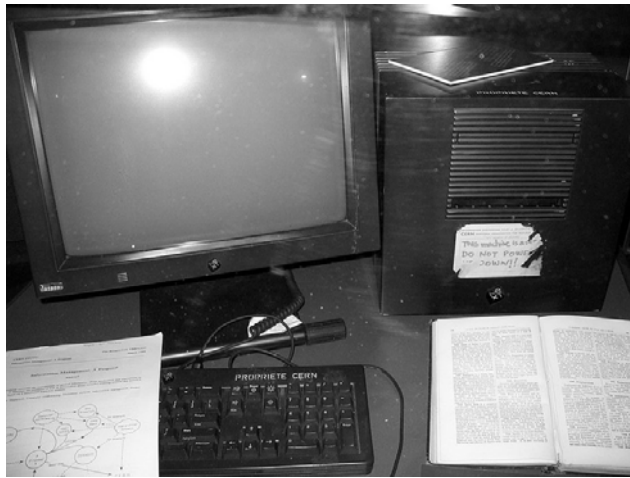
⁶⁹ http://de.wikipedia.org/wiki/Chronologie_des_Internets

⁷⁰ <http://de.wikipedia.org/wiki/Internet>

⁷¹ Siehe: <http://www.w3.org/People/Berners-Lee>

⁷² Conseil Européen pour la Recherche Nucléaire.

eine Patentierung zu verzichten, da dies den durchschlagenden Erfolg der Idee wenn schon nicht verhindert, doch aber zumindest stark verzögert hätte.



Der erste Webserver in CERN
http://en.wikipedia.org/wiki/Image:First_Web_Server.jpg

Einige wesentliche Aspekte des WWW sind:

Informationsintegration

Dies bedeutet, dass alle Informationen aufgenommen, zu anderen hinzugefügt und verknüpft werden können. Von der potentiell universellen Verknüpfbarkeit aller Datenknoten rührt auch der Gebrauch Metapher ‚Netz‘ her.

Medienintegration

Weltweit wachsen alle Informationen zusammen.

Offenes System

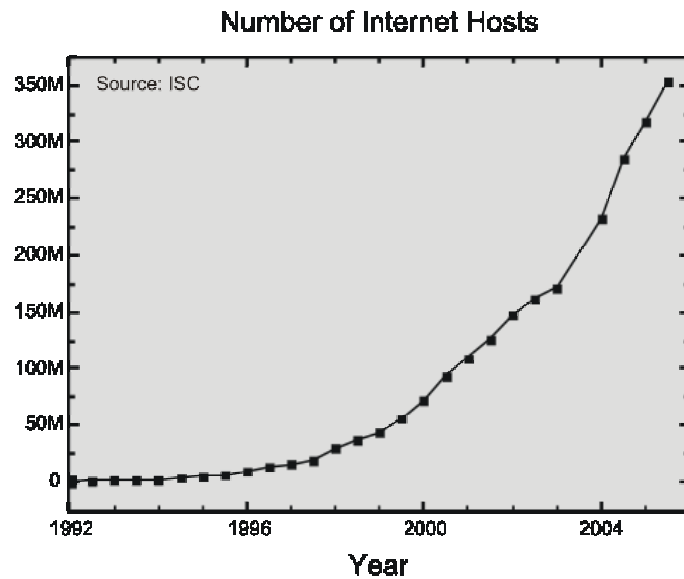
Das WWW ist potentiell unendlich erweiterbar.

Demokratie

Von der Grundkonzeption her verstanden, ist das Internet als demokratisches Medium zu sehen, in dem es keine Zensur gibt und jeder seine Meinung frei äußern kann und jeder seinen Beitrag zum WWW leisten kann, in dem er zum Beispiel Informationen hinzufügt. Es zeigt sich jedoch in den letzten Jahren, dass diese Eigenschaft nicht überall zutrifft. So gibt es in manchen Ländern rechtliche Beschränkungen, bestimmte Themen zu veröffentlichen und auch viele Server oder Dienste sind an Auflagen gebunden.

Global und integrativ

Es spielt keine Rolle, wo die Information zur Verfügung gestellt wird.



Die Anzahl von Internetuser von 1992 bis 2006 in Millionen
http://en.wikipedia.org/wiki/Image:Number_of_internet_hosts.svg

5.6.3 Web-Based-Training (WBT)

Mit der zunehmenden Verbreitung des WWW und der damit einhergehenden Vernetzung von Computern wurde in vielen Bereichen des *E-Learnings* der Schritt vom Computer basierten Lernen (CBL) hin zum Web-basierten Lernen (WBL) gemacht. Die Zahl von Autorensystemen und Courseware stieg zwar stetig, aber gleichzeitig sank die Qualität der Produkte stark, was oftmals der Reputation der Firmen und der von Courseware sowie dem WBT im Allgemeinen abträglich war.⁷³

Der große Vorteil den WBT bietet ist, dass Lehrer und Lernende sich nicht an einem Ort befinden müssen und somit ganz neue pädagogische Modelle denkbar werden.

Man unterscheidet zwischen drei Arten des WBT:

Open Distance Learning

Diese Art des WBT ist ein klassischer Fernlehrcurs. Die Lernmaterialien werden vom Lehrenden für den Lernenden zusammengestellt und übermittelt. Der Tutor steht dem Lernenden für Anfragen und Hilfestellungen bei Problemen zur Verfügung, das bedeutet, dass auf individuelle Bedürfnisse von Lernenden eingegangen werden kann. Zur Kommunikation können E-Mail, Chats und Newsgroups verwendet werden.

Teletutoring

Hier werden über Videokonferenzen, die Kommunikation in Echtzeit bieten oder Online-Seminare in virtuellen Klassenzimmern, Gruppen von Lernenden betreut.

⁷³ Holzinger 2001, Bd. 2, S. 205.

Diese Art von WBT steckt noch in den Kinderschuhen, da sie verhältnismäßig ressourcenintensiv ist und hohe Bandbreiten braucht.

Teleteaching

Dies ist eine Anlehnung an das Telelernen in den 50er und 60er Jahren. Der Lernende verhält sich passiv und es ist keine Kommunikation unter den Teilnehmern vorgesehen. Die häufigsten Varianten von Teleteaching ist, die Übertragung einer Vorlesung in Seminarräume via Videokonferenz oder Lernende bearbeiten Vorlesungen anhand von Texten oder multimedialen Medien, die im Web zugänglich sind.

Das WBT bietet gegenüber seinem Vorgänger dem CBL eine ganze Reihe evidenter Vorteile⁷⁴:

- Die Aufgaben und Lernmaterialien können dem User leicht und kostengünstig übermittelt werden.
- Die Informationen können schnell an sich verändernde Verhältnisse adaptiert werden. So können, zum Beispiel neue wissenschaftliche Erkenntnisse schnell weitergegeben werden.
- Die Betreuung von Gruppen ist genauso möglich, wie die von einzelnen Lernenden.

5.6.4 Distance Learning: Virtuelle Universitäten

Bildungseinrichtungen müssen heutzutage oft mit finanziellen Einsparungen von Seiten der Regierung leben. Dies erschwert ihnen die Möglichkeiten neue Medien und Technologien ihren Schülern vorzustellen und sie im Umgang mit Informationstechnologie zu schulen.⁷⁵ Gleichzeitig bieten aber gerade diese neuen Technologien ein enormes Einsparungspotential. Denn die Lehrveranstaltungen könnten auch über das Internet angeboten werden. Dies erspart den Bildungseinrichtungen einerseits Räumlichkeiten und bringt so mehr Effizienz. So ist der Zugang zu Vorlesungen oder Lehrmaterial orts- und zeitunabhängig. Durch die neuen Medien können diese auch zwischen Universitäten ausgetauscht werden, um ein größeres Kursangebot anbieten zu können.

Auch können internationale Erfahrungen gesammelt werden, ohne Auslandsaufenthalte zu absolvieren. Diese Möglichkeiten sind natürlich nicht nur auf

⁷⁴ http://de.wikipedia.org/wiki/Web_Based_Training

⁷⁵ Holzinger 2001, Bd. 2, S. 207.

den universitären Bereich beschränkt, vielmehr treffen die ausgeführten Beobachtungen auf den gesamten Bildungssektor zu.

Am Ende dieser Arbeit sei damit noch auf eine immer weiter verbreitete Lernmethode, das so genannte *blended learning* hingewiesen, bei dem konventionelle Lehrmethoden mit den Möglichkeiten, die Computer und Internet bieten, kombiniert werden. Hier können zum Beispiel die Vorzüge des Präsenzunterrichts mit denen des Fernunterrichts kombiniert werden. Die intensivere Kommunikation durch eine größere Zahl an Rezipienten und die Austauschbarkeit von Materialien führt zweifelsohne zu mehr Qualität, da Lernende beobachten können, wie an anderen Schulen oder in anderen Ländern der Unterricht geführt wird. Dies könnte in Zukunft zu mehr Konkurrenz zwischen Bildungseinrichtungen führen. Letztlich wäre eine Globalisierung in diesem Bereich mit besserem Zugang zu Bildung eine wünschenswerte Entwicklung.

6 Unterrichtsorganisation mit Hilfe von Content Management Systems

Die Lernumgebung *Moodle* kommt im Zuge des Projekts *Aims* an unserer Schule bereits in zahlreichen Projekten zum Einsatz. Neben *Moodle* gibt es natürlich eine Vielzahl von anderen kommerziellen Content Management Systems. Die am weitest verbreiteten sind *Blackboard* und *WebCT*⁷⁶. Bekannte Open-Source-Produkte sind abgesehen von *Moodle Bazaar* und *Dokeos*.

Im folgenden Abschnitt werde ich versuchen, allgemeine Informationen zu *Moodle* zusammenzustellen, und danach auch auf einige konkrete Projekte eingehen.

Mit der exponentiell ansteigenden Zahl an digitalen Daten, die weltweit produziert werden, stellt sich in zunehmendem Maße das Problem der Organisation dieser Daten. Eine wichtige Rolle spielen hierbei die bereits erwähnten Content Management Systeme, das sind Programme, die die gemeinschaftliche Erstellung, Bearbeitung und Nutzung von multimedialen Inhalten organisieren. Eine besondere Form derartiger Redaktionssysteme, die im Schul- und Universitätsbereich immer größere Bedeutung erlangen, stellen Learning Management Systems dar. Für diese Art von Software ist eine ganze Reihe unterschiedlicher Termini gebräuchlich. Neben Learning Management Systems und Course Management Systems spricht man auch von Managed Learning Environments oder Virtual Learning Environments, in ähnlicher Bedeutung wird aber auch Computer-mediated Communication gebraucht. *Moodle* zählt zu den so genannten Course Management Systems, die wie der Name schon sagt, nicht wie Content-Management Systeme die Aufgabe der Präsentation von Inhalten im Allgemeinen haben, sondern Inhalte für Kurse aufbereiten sollen.

6.1 Course Management Systems (CMS)

CMS sind heute in der Regel web-basierte Applikationen, die einerseits den Lehrenden eine Vielzahl von Tools zur Verfügung stellen, um Kurse zusammenzustellen, und andererseits den autorisierten Lernenden Zugang zu diesen bieten.

Zu den wichtigsten Funktionen, die ein derartiges System zu erfüllen hat, gehören:

Dateien auf einen Server hochladen und anderen zur Verfügung stellen

⁷⁶ Im Oktober 2005 gaben Blackboard und WebCT bekannt, dass sie fusionieren und unter dem Namen Blackboard weiterbestehen würden.

Anstatt ein HTML-Dokument zu erstellen und via FTP auf einen Server zu stellen, bieten viel CMS die Möglichkeit, Dokumente über Online-Formulare zu veröffentlichen.

Foren und Chats

Diese Art der Kommunikation bietet Lernenden virtuelle Treffpunkte, um zu diskutieren. Foren haben zusätzlich den Vorteil, dass die Qualität des Gesprächs durch die Möglichkeit steigt, sich für seine Fragen, Argumente und Antworten länger Zeit zum Überlegen lassen zu können. Chats ermöglichen hingegen eine Kommunikation, bei der man nicht lange auf die Antwort warten muss und daher schnell und einfach seine Gedanken austauschen kann.

So können sich Lernende besser präsentieren, als sie dazu im Klassenverband vielleicht in der Lage wären. Auch können exakter formulierte Fragen gestellt werden, die einerseits den Lehrern die Gelegenheit bietet, diese ausführlich zu beantworten, und die andererseits der ganzen Gruppe von Lernenden zur Verfügung gestellt werden können.

Quiz und Wissensüberprüfungen

Der Lernerfolg kann sofort überprüft werden, wodurch die Lernenden schnelles Feedback erhalten. So kann es hilfreich sein, nach durchgenommenen Kapiteln eines Buches einen Fragekatalog angeboten zu bekommen.

Sammeln und Korrigieren von Aufgaben

Aufgaben von Schülern online zu erhalten vereinfacht den Prozess des Korrigierens enorm. Auch lassen sich interaktive Aufgaben daraus machen, indem man Schüler veranlasst, die anonymisierten Aufgaben ihrer Mitschüler zu kontrollieren. Dies kann auch im Hinblick auf Motivation hilfreich sein.

Benotung

Für den Lernenden besteht jederzeit die Möglichkeit, sich über seinen aktuellen Notenstand zu informieren. Der Schüler ist daher immer über seine Leistungen auf dem Laufenden, was einerseits als Ansporn und Belobigung, andererseits als Warnung dienen kann.

6.1.1 Moodle

Moodle ist eine multifunktionale sehr flexibel einsetzbare Lernumgebung, die Lernende unterstützt, sich online Wissen anzueignen. Es hat keine inhaltlichen Beschränkungen und die Kursräume können von den Verfassern der Kurse nach

Belieben mit eigenen Inhalten gefüllt werden. Die verhältnismäßig benutzerfreundliche Oberfläche ermöglicht sogar Personen die Kurserstellung, die über keine Programmierkenntnisse verfügen und im Umgang mit digitalen Instrumenten nicht so versiert sind.

6.1.1.1 Allgemeines

Ursprünglich stand *MOODLE* für *Martin's Object-Oriented Dynamic Learning Environment*. Später wurde das *M* aus dem Vornamen des Entwicklers Martin Dougiamas uminterpretiert, weshalb die Abkürzung nun für *Modular Object-Oriented Dynamic Learning Environment* steht.

Moodle wurde in insgesamt 61 verschiedene Sprachen übersetzt. Die Entwicklung an *Moodle* begann bereits 1999, durch seine Arbeit bei WebCT und seine Doktorarbeit in Philosophie war Martin Dougiamas in der Lage, eine komplett neue Lernumgebung auf interdisziplinärer Basis zu schaffen.

6.1.1.2 Technische Eckdaten

Moodle ist modular aufgebaut, das bedeutet, dass es sich aus mehreren Teilprogrammen, so genannten Modulen, zusammensetzt.

Es ist grundsätzlich für das Internet konzipiert und baut daher vornehmlich auf HTML auf. Es lassen sich aber Lerninhalte in vielen anderen Formaten zum Beispiel PDF oder Macromedia Flash in die Kurse integrieren.

Bemerkenswert ist auch, dass es plattformunabhängig ist, das bedeutet, dass es unter Windows, Unix, Linux, Mac OS und allen anderen Plattformen, die die Skriptsprache PHP unterstützen, problemlos installiert werden kann. Die wichtigsten Datenbanksysteme MySQL und PostgreSQL lassen sich mit *Moodle* ebenso kombinieren wie andere kommerzielle Produkte.

6.1.1.3 Open Source

Moodle ist *open source*. Die Grundidee von *open source* ist die Freiheit und Weitergabe von Wissen. Im Softwarebereich bedeutet dies, dass der Quellcode öffentlich zugänglich ist und ohne Lizenzgebühren kopiert, verändert und weitergegeben werden darf. *Moodle* kann somit von jedem Interessierten gratis heruntergeladen und benutzt werden.

6.1.1.4 Lerntheoretische Grundlage

Im Gegensatz zu anderen CMS wurde *Moodle* auf dem Konzept des konstruktivistischen Lernens aufgebaut. Demnach stützt sich *Moodle* auf bereits vorhandenes Wissen und Lernerfahrungen, die Lernenden eignen sich Wissen im Kontakt mit anderen an. So baut *Moodle* darauf auf, dass man lernt indem man sein Wissen anderen präsentiert.

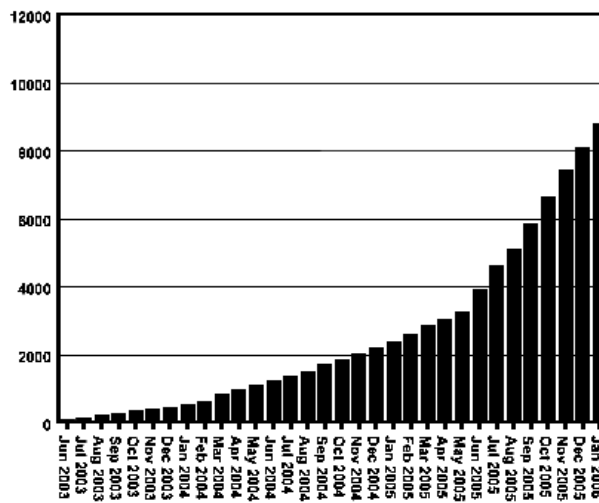
Während andere CMS die Lehrenden dazu auffordern statisches Material hochladen zu lassen, fokussiert *Moodle* Diskussionen und auf das Zur-Verfügung-stellen von Informationen durch die User selbst.

6.1.1.5 Community

Um *Moodle* hat sich eine große aktive Community gebildet, die dazu beiträgt, dass das System ausgebaut und weiterentwickelt wird. Die *Moodle* Community zählt laut <http://moodle.org/stats> 90 500 Mitglieder und über 9 000 *Moodle* Webseiten. Sie war ohne Zweifel mitverantwortlich für den Erfolg der Plattform. Eine Anwendergruppe dieser Größe eröffnet die Möglichkeit, Antworten und Hilfestellungen zu finden. Auch verhindert sie, dass Tools, die noch nicht genug ausgereift sind, allzu große Verbreitung finden und schon gar nicht in die offizielle Distribution gelangen. Dies trägt zu Qualitätssicherung bei, für die man bei kommerziellen Produkten normalerweise Testpersonen engagieren müsste.

Moodle sites

Total known sites



Moodle 1.5 sites: 6180 (of 9321)
 Courses: 244217
 Users: 2607334
 Teachers: 436909
 Enrolments: 3776106
 Forum Posts: 2752338
 Resources: 1489464
 Quiz questions: 1661708

Die Anzahl der Moodleseiten, der Kurse und der Benutzer
<http://moodle.org/stats/>

6.1.1.6 Die Erstellung eines Kurses in Moodle

Moodle bietet nicht nur für Lernende einen einfachen Einstieg. Auch Lehrenden und Tutoren haben die Möglichkeit ohne viel Know-how und Zeitaufwand auf einer benutzerfreundlichen Plattform Kurse zu erstellen, die genug Platz zur Entfaltung individueller Vorstellungen bietet.

Die Erstellung eines Kurses erfolgt über ein Formular. Hier werden bereits der Kursname, eine Kurzpräsentation des Themas und einige weitere Eigenschaften festgelegt, wie zum Beispiel der Teilnahmezeitraum.

Für den Verfasser eröffnen sich mit einem neuen Kursprojekt zahlreiche Möglichkeiten zur Gestaltung des Kurses, auf die ich im Folgenden näher eingehen möchte. Der Kursersteller hat die Auswahl zwischen drei Formaten:

Wochen-Format

Man kann einen fixen Start- und Endtermin des Kurses bestimmen in dessen zeitlichem Rahmen er stattfindet. Jede Woche werden neue Aktivitäten angeboten, manche sind nur für einen bestimmten Zeitraum verfügbar.

Themen-Format

Der Kurs wird anstatt in zeitliche in thematische Abschnitte gegliedert und man benötigt keinen fixen Start- und Endtermin.

Soziales Format

Im Zentrum eines solchen Kurses steht das „soziale Forum“, auf dem der Kurs aufbaut und in dem die Teilnehmer mehr Freiräume haben. Man kann das soziale Format auch einfach nur für Diskussionen nutzen.

6.1.2 Projekt *Aims*

Das Bundesministerium für Bildung, Wissenschaft und Kunst hat im Schuljahr 2004/2005 in Zusammenarbeit mit dem Pädagogischen Institut Wien das Projekt *Aims* initiiert. Das Ziel des Projekts ist die Verankerung eines Informationsmanagementsystems an Schulen. Die Schulbibliothek soll hierbei als multimediales Zentrum fungieren. Derzeit sind im Rahmen dieses Projekts acht Wiener Gymnasien über Lern- und Informationsplattformen miteinander verbunden und nach dreijähriger Testphase sollen die Ergebnisse an der Universität Wien ausgewertet werden.

Unsere Schule befindet sich nunmehr im 2. Projektjahr und es lässt sich beobachten, dass die neue Lernplattform bei Schülern größerer Beliebtheit erfreut als Vorgängerprojekte. Eine Hürde vor der sowohl Lehrkräfte als auch Schüler stehen, ist sich auf ein neues Unterrichtskonzept einzulassen und mehr auf Projekt- und Gruppenarbeit zu setzen.

Ziele des Projekts sind:

- selbstständige Wissensaneignung
- Gewöhnung an die Lernumgebung
- neue Lern- und Lehrmethoden
- Schul- und Unterrichtsorganisation

Das heißt, dass besonders individualisiertes Lernen im Vordergrund stehen sollte. Die Förderung von Teamarbeit stellt ebenfalls ein wichtiges Ziel dar. Es sollen Rahmenbedingungen zur Schaffung von Strukturen ermöglicht werden, die zur effektiveren Kooperation beitragen und auch schulübergreifend einsetzbar sind.

Bei dem Projekt *Aims* sollten die verschiedenen Möglichkeiten und Aspekte des E-Learnings gemeinsam erarbeitet bzw. dem Wissensaustausch unter den Schülern ein hoher Stellenwert eingeräumt werden.

6.1.2.1 Astrologieprojekt 2005

Im Dezember 2005 wurde die Lernplattform *Moodle* für ein viertägiges fächerübergreifendes Astrologieprojekt einer achten Klasse herangezogen. Nicht nur für die meisten Schüler, sondern auch für die Professoren war dies eine völlig neue Erfahrung.

So erläuterte mir Herr Professor Stehle in einem Gespräch im Jänner 2006 einige interessante Beobachtungen, die er im Laufe des Projekts machen konnte. Seiner Meinung nach eröffne diese Lernplattform „großartige Möglichkeiten“ jedoch stoße man immer wieder auf Anlaufschwierigkeiten, die leicht behoben werden könnten. Der Aspekt, dass *Moodle* alles genau dokumentiere und Schüler bei Multiplechoicetests gleich beurteilt werden, mache den Unterricht transparenter und liefere auch den Schülern die Möglichkeit, sich immer über ihren derzeitigen Notenstand zu informieren. Als einen Nachteil sieht er, dass die Erstellung der Kurse sehr zeitaufwändig ist. Durch die Wiederverwendbarkeit der Materialien gleiche sich dies jedoch wieder aus und Verbesserungen können in neuen Versionen leicht angefertigt werden. Im Laufe des Gesprächs warnte Professor Stehle auch davor das Medium zu überschätzen und sich zu sehr auf die Technologie selbst zu konzentrieren. Dies könne dazu führen, dass der Inhalt zu kurz komme. In Bezug auf das konkrete Projekt meinte er, dass von Seiten der Schüler nur positive Rückmeldungen gekommen seien, auch machte er die Beobachtung, dass sich alle aktiv an dem Projekt beteiligt und großes Interesse gezeigt haben, was auf die öffentliche Präsentation zurückzuführen ist, die den Abschluss des Projekts bildete. Sehr wichtig war für Herrn Professor Stehle, die Plattform nicht nur als Erweiterung zum Unterricht zu sehen, sondern auch als Kommunikationsmittel unter den Lehrern, wodurch die Koordination sehr vereinfacht würde.

6.1.2.2 Lateinprojekt

In einem Gespräch mit Frau Professor Pacher, das ich ebenfalls im Jänner 2006 führte, waren in Bezug auf ihre Erfahrungen mit *Moodle* ebenfalls Schlagwörter wie „umständlich und zeitaufwändig“ zu hören. Sie löste dieses Problem, indem sie zusätzliche Software hinzuzog, mit der sie in der Lage war, Kreuzworträtsel⁷⁷ und Lückentexte⁷⁸ mit geringerem Aufwand zu erstellen. Sie ist der Meinung, dass diese

⁷⁷ Crossword Compiler.

⁷⁸ Hotpotatoes.

Plattform besonders für die Unterstufe geeignet ist, da besonders jüngere Schüler durch die spielerische Komponente motiviert werden könnten. Konkret auf das Fach Latein bezogen bietet *Moodle* die Möglichkeit, vor Schularbeiten zusätzliche Übungen freiwillig zu erhalten. Bei Entfall von Stunden stellt sie Arbeitsaufträge auf die *Moodle* Plattform. Auf die Frage wie das Feedback von Schülern gewesen ist, meint sie, dass die Motivation groß war und die Plattform stark frequentiert wurde, nachdem die ersten Aufgaben und Übungen online waren. Abseits von *Moodle* hat Frau Professor Pacher bereits vor der Einführung dieser Lernplattform an unserer Schule begonnen, Schüler, die Hilfe benötigen, per E-Mail Betreuung anzubieten.

7 Schlussbetrachtung: Die Zukunft des Lernens?

In viele neue Technologien und Innovation wurden in den letzten Jahren zu hohe Erwartungen gesetzt, die aus Stimmungen und Strömungen in der Wirtschaft und in der Bevölkerung hervorgingen. Auch die „eLearning-Hype“⁷⁹ wurde durch die Erkenntnis, dass herkömmliches Lernen nicht zur Gänze durch Virtualisierung der Lernprozesse ersetzt werden kann, abgeschwächt. Dies führte dazu, dass methodisch hybride Ansätze, Kombinationen aus herkömmlichem Lernen und E-Learning größere Popularität fanden. In diesem Zusammenhang spricht man heute häufig auch vom so genannten blended learning. Die Schüler besuchen dabei nach wie vor konventionellen Unterricht, der aber durch digitale Materialien und computerbeziehungsweise netz-basierte Methoden unterstützt wird.

Oft werden CMS heute als Hybrid Kurse verwendet. Das heißt, dass sowohl die Vorteile des unmittelbaren Unterrichts in der realen als auch der virtuellen Welt in den Unterricht einbezogen werden, um einen maximalen Lernerfolg zu erzielen. So bewährte es sich oftmals, reine Informationsvermittlung online abzuwickeln, wohingegen sekundäre Problembewältigung effektiver im direkten Lehrer-Schüler-Kontakt abzuwickeln sind.

Allgemein kann man auch feststellen, dass die Technologie noch lange nicht im Stande ist, das Lernen vollständig im virtuellen Raum stattfinden zu lassen. Zu diesem Schritt sind weder die Lernenden bereit, noch sind virtuelle Tutoren und Klassenzimmer weit genug entwickelt, um ihre realen Gegenstücke zu ersetzen. Ohne jeden Zweifel ist das E-Learning aber auf seinem Vormarsch nicht aufzuhalten, wenn auch das Tempo, mit dem es Verbreitung findet, hinter den ursprünglichen Erwartungen zurückbleibt. Zu hoffen bleibt dabei, dass die Verantwortlichen auf allen Ebenen in Zukunft vermehrt in innovatives Lernen investieren.

⁷⁹ Reinmann-Rothmeier 2003, S. 28.

Glossar

Algorithmus

ist ein häufig in der Informationstechnologie verwendeter Terminus; er ist eine exakt definierte Handlungsanweisung, die die Lösung eines spezifischen Problems oder auch einer ganzen Problemgruppe ermöglicht.

Blended Learning

Übersetzt: „vermisches Lernen“; auch *hybrides Lernen* genannt; man versteht darunter eine Kombination aus herkömmlichen pädagogischen Ansätzen mit Methoden des E-Learnings.

FTP

Das File Transfer Protocol (engl. für Datenübertragungsverfahren) ist ein Netzwerkprotokoll, mit dessen Hilfe Dateien in Netzwerken ausgetauscht werden.

HTML

Abkürzung für Hypertext Markup Language; Sprache zur Auszeichnung von Hypertext-Dokumenten zur Verwendung im World Wide Web; von Tim Berners-Lee in CERN initiiert.

Hyperlinks

Verweis von einem Element eines Hypertextdokuments zu einem anderen; jedes Link enthält die Adresse seines Ziels; sind in HTML üblicherweise unidirektional⁸⁰.

Hypermedia

Enthält ein Dokument neben Text auch Audioelemente und visuelle Komponenten, so spricht man von Hypermedia.

Hypertext-System

Software zur Speicherung und inhaltlichen Verknüpfung von Dokumenten; nicht-lineare Struktur; das Wort Hypertext entstand während der Arbeit am Projekt XANADU⁸¹.

Knoten

Informationseinheiten von Hypertext, die über Links nicht linear miteinander verbunden sind; können sowohl Texte als auch Grafik- oder Bildelemente sein.

Online

bedeutet, dass ein Computer mit einem anderen Gerät oder einem Netzwerk verbunden ist.

Open Source Software

bedeutet, dass der Quellcode eines Programms allen potentiellen Anwendern frei zur Verfügung steht; diese dürfen ihn ohne Lizenzen zu bezahlen verwenden und auch weiterentwickeln; kann zur schnellen Weiterentwicklung eines Programms beitragen, oft können Fehler schnell und effektiv behoben werden.

PHP

Skriptsprache zur Erstellung dynamischer Websites.

⁸⁰ Das Ziel weiß nichts von der Linksetzung.

⁸¹ Holzinger 2001, Bd. 2, S. 187. Siehe auch: http://en.wikipedia.org/wiki/Project_Xanadu

Literaturverzeichnis

- Claus, Volker / Schwill, Andreas*: Informatik (Schülerduden). Mannheim ³1997.
- Cole, Jason*: Using Moodle. Teaching with the Popular Open Source Management System. Sebastopol 2005.
- Eder, Franz X. / Fuchs, Eduard*: Webbasiertes historisches Lehren und Lernen am Beispiel *Geschichte Online*. In: Historische Sozialkunde 3/2005.
- Holzinger, Andreas*: Basiswissen Multimedia. Band 2: Lernen. Würzburg 2001.
- Holzinger, Andreas*: Basiswissen Multimedia. Band 3: Design. Würzburg 2001.
- Kuhlen, Rainer*: Hypertext. Ein nicht-lineares Medium zwischen Buch und Wissensbank. Berlin 1991.
- Martens, Alke*: Ein Tutoring Prozess Modell. Hildesheim 1970 (Dissertation).
- Niegemann, Helmut M.* Neue Lernmedien konzipieren, entwickeln, einsetzen. Bern 2002.
- Reinmann-Rothmeier, Gabi*: Didaktische Innovation durch Blended Learning. Leitlinie anhand eines Beispiels aus der Hochschule. Bern 2003.
- Scheffer, Ute / Hesse, Friedrich W. (HG.)*: E-Learning. Die Revolution des Lernens gewinnbringend einsetzen. Stuttgart 2002.
- Schulmeister, Rolf*: Grundlagen Hypermedialer Lernsysteme. Theorie-Didaktik-Design. München 2002.
- Skinner, B. F.*: Teaching Machines. Reprinted from SCIENCE 1958, Vol. 128. (<http://www.bfskinner.org>).

Webseiten

<http://ei.cs.vt.edu/~history/GASCH.KAY.HTML>

<http://kelty.rice.edu/375/lectures/bush>

http://www.artmuseum.net/w2vr/archives/Kay/01_Dynabook.html

<http://www.bfskinner.org>

<http://www.moodle.org>

<http://www.papert.org>

<http://www.parc.xerox.com>

<http://www.w3.org/People/Berners-Lee>

<http://www.wikipedia.org>

<http://xanadu.com.au/ted>

Autor: Ruben Merlin MÖRTH

Geboren: 10.03.1988

Adresse: Belvederegasse 8/3/10, 1040 Wien

ERKLÄRUNG

Hiermit erkläre ich, dass ich die vorliegende Fachbereichsarbeit selbst verfasste und nur die angegebenen Quellen verwendete.

Wien, 17. Februar 2006

Ruben Merlin Mörth

Protokoll

zur Fachbereichsarbeit im Wahlpflichtfach Informatik

Zukunft des Lernens – vom Computer-Based-Training zum Web-Based-Training

Verfasser: Ruben Merlin MÖRTH
Betreuer: Mag. Carl METNITZ

GRG1 Stubenbastei, Stubenbastei 6-8
Schuljahr 2005/2006

Juni 2005	Ich einige mich mit Herrn Mag. Metnitz auf den Inhalt meiner Arbeit und erhalte die ersten Literaturtipps.
Juli-August 2005	Ich beginne mich mit dem Thema auseinander, beschaffe mir die Literatur und überlege mir den Aufbau meiner Arbeit.
September 2005	Am Anfang des Monats stelle ich Herrn Mag. Metnitz meine genaueren Pläne vor und wir legen den Titel der Arbeit fest, woraufhin die Grobdisposition eingereicht wird.
Oktober 2005	Ich beginne mich einzulesen und suche in Bibliotheken nach weiteren Quellen, die ich in der Arbeit verwenden könnte.
November 2005	Das erste Kapitel hat das Thema „Lerntheorien“ zum Inhalt.
Dezember 2005	Nach regelmäßigen Besprechungen mit Herrn Mag. Metnitz beginne ich am Monatsanfang das zweite Kapitel, das ich auch bald darauf fertig stelle. Mitte Dezember beginne ich das dritte Kapitel, unterbreche die Arbeit daran aber in den Weihnachtsferien, um mich mit dem umfangreicheren fünften Kapitel zu befassen. Zur gleichen Zeit arbeite ich die ersten Korrekturen von Herrn Prof. Metnitz ein.
Jänner 2006	Nach den Ferien schließe ich zuerst das bereits begonnene 3. Kapitel ab, um dann nach weiterer Durchsicht meines Betreuers mein letztes und wichtigstes Kapitel anzufangen. Zur Arbeit an diesem Kapitel bekomme ich zusätzliche Projektunterlagen und arbeite anhand einer neuen englischen Publikation an diesem Kapitel. Ende Jänner führe ich noch zusätzlich zwei Gespräche mit Lehrern, damit deren Erfahrungen mit dieser Lernplattform einfließen können.
Februar 2006	In den Semesterferien erhalten meine Arbeit und besonders mein letztes Kapitel den letzten Schliff, ich verfasse die Einleitung und meine Schlussbetrachtungen.
17. Februar 2006	Abgabe der Fachbereichsarbeit.

Wien, 17. Februar 2006

Ruben Merlin Mörth