

**Fachhochschule Braunschweig / Wolfenbüttel**

# **Entwicklung eines Mathematik – Lernprogramms für Kinder**

## **Masterarbeit**

Im Weiterbildungsstudiengang Multimedia

Von

**Nguyen Nam Khanh**

Matrikelnummer: 40587745

Betreuer: **Prof. Dr. Helmut Voullieme**

**Dipl. Soz. Päd. Klaus Henning**

**Dipl. Des. Berit Andronis**

Abgabe der Arbeit: **15.06.2008**

## **Danksagung**

Hier möchte ich mich bei all jenen bedanken, die mich während der Arbeit an dieser Masterarbeit unterstützt haben. Zuerst möchte ich mich bei Herrn Dipl. Soz. Päd. Klaus Henning, Herrn Prof. Dr. Helmut Voullieme und Frau Dipl. Des. Berit Andronis für die Betreuung bedanken. Und noch vielen Dank an meine Eltern und Bac Kiet für ihre Unterstützung meines gesamten Studiums.

## Inhaltsverzeichnis

<b>1. Einleitung</b> .....	<b>4</b>
1.1. Hintergrund zur Programmidee.....	5
1.2. Lernaktivitäten .....	6
1.3. Die Nutzung von Lernprogramm durch Kinder.....	7
<b>2. Beschreibung des Programms</b> .....	<b>8</b>
2.1. Art des Programms.....	8
2.2. Lektionen und Themen.....	9
2.3. Einsatzorte .....	10
2.4. Hardware- und Softwarevoraussetzungen.....	10
<b>3. Didaktik</b> .....	<b>12</b>
3.1. Zielgruppe .....	12
3.2. Lehrziele und Lehrinhalte .....	13
3.2.1. Lehrziele .....	13
3.2.2. Lehrinhalte und Struktur eines Inhaltsbereichs .....	15
3.2.3. Struktur einer Aufgabe.....	15
3.2.4. Aufgabentypen.....	17
3.3. Lernmotivation .....	19
<b>4. Gestaltung</b> .....	<b>24</b>
4.1. Screendesign.....	24
4.1.1. Startseite.....	24
4.1.2. Arbeitsseite .....	26
4.2. Navigation .....	29
<b>5. Technische Entwicklung</b> .....	<b>32</b>
5.1. Macromedia Flash .....	32
5.2. Programmierung.....	35
5.2.1. Animationen.....	35
5.2.2. Loadmovie .....	36
5.2.3. Lückentexte.....	36
5.2.4. Multiple-Choice .....	37
<b>6. Zusammenfassung und Ausblick</b> .....	<b>41</b>
<b>7. Verzeichnis</b> .....	<b>43</b>
7.1. Abbildungsverzeichnis .....	43
7.2. Literaturverzeichnis.....	43

## 1. Einleitung

Der Computer ist heutzutage ein wichtiger Teil im Leben von Kinder und Jugendliche. Kinder und Jugendliche verbringen schon viele Stunden am Computer. Mit dem Computer können sie Internet surfen, chatten, spielen und lernen. Die rasante Entwicklung der Computertechnologien ebnete nicht nur den Weg für neue Arten der Kommunikation, sondern auch für eine neue Art des Lernens. Doch warum ist heute der Computer ein geeignetes Medium um Wissen zu vermitteln?

Es gibt immer viele Vorteile, wenn man beim Computer mit einer Lernsoftware lernt. Ein großer Vorteil für den Lerner ist die Individualisierung des Lernens. Der Lerner kann selbst bestimmen, welche Lerninhalte für ihn wichtig sind und wie intensiv er diese lernen möchte. Außerdem hat jeder Lerner sein eigenes Lerntempo. In der Schule wird Lehrer bei der Vermittlung von neuen Inhalten ein bestimmtes Tempo vorgeben, welches für den Großteil der Schüler adäquat ist. Nicht jeder kann diesem Tempo folgen. Aber mit einer Lernsoftware kann der Lerner das Lerntempo selbst bestimmen. Dadurch hat der Lerner die Möglichkeit, bestimmte Übungen oder Lektionen beliebig oft zu wiederholen, bis er sich selbst sicher fühlt.

Auch den Ort des Lernens kann der Lernende selbst nach Belieben auswählen und kann auf diese Weise Geld und Zeit sparen. Wenn man Lust und Zeit hat, kann das Programm genutzt werden. Es sind keine langen Wege zur Schule oder Ausbildungsstelle mehr nötig und es kann bequem zum Beispiel zu Hause vor dem Computer gelernt werden.

Ein weiterer Vorteil für den Lerner ist die Vielfältigkeit und der Abwechslungsreichtum des Lernens mit neuen Medien. Ein Lernprogramm mit farbigen Animationen und interaktiven Elementen ist jedenfalls fesselnder als die herkömmlichen Lernmethoden, wie zum Beispiel einfache gedruckte Texte oder auch Bücher. Der Lernende gibt Antworten, ergänzt und wählt aus. Der Computer reagiert auf jede Aktion. Lernsoftware fördert aktives statt passives Wissen. Auch allein die Benutzung eines Computers bereitet vielen Lernern Freude, besonders

denjenigen, die nicht daran gewöhnt sind, mit einem PC zu lernen oder zu arbeiten (zum Beispiel Kinder und Jugendliche). Sie sind meist schon sehr stolz darauf, dieses für sie neue und ungewohnte Medium zu beherrschen und zu bewältigen, was natürlich die Lernmotivation gewaltig steigert.

Außerdem mit wachsender Leistungsfähigkeit der Rechner, umfangreicher Bandbreite der unterschiedlichen Hard- und Softwarelösungen sind viele Formen des E-Learning entstanden, zum Beispiel: Spielekonsole, Vorlesungen und Übungen mit Anleitung können über das Internet übertragen werden. Handys sind auch ein gutes Mittel um Wissen zu vermitteln... Mit vielen Vorteilen und mit der Hilfe von neuen Technologien ist es selbstverständlich, dass die Akzeptanz von Lernprogramm immer größer wird. Zurzeit findet man Lernprogramm überall, in der betrieblichen Aus- und Weiterbildung, in Schulen, in Universitäten und auch im privaten Bereich.

Für meine Masterarbeit habe ich ein Lernprogramm erstellt, das für Kinder in 3.Klasse erstellt wurde. Dieses Programm beinhaltet eine Lernwelt mit verschiedenen Lernmethoden und Spielelementen, die durch die mediale Unterstützung des Computers eine nette Lern- und Spielwelt bieten.

### **1.1. Hintergrund zur Programmidee**

Lernprogramm für Kinder ist für mich ein sehr interessantes Thema. Ich habe mich für die Mathematik als Lernziel für dieses Programm aus mehreren Gründen entschieden.

- Ich glaube, dass Mathematik ein trockenes Fach ist. Kinder können normalerweise nicht zu lang beim Schreibtisch sitzen um die Matheaufgaben zu üben. Aber mit einem spielerischen Lernprogramm hoffe ich, dass Kinder mehr Spaß an Mathematik haben würden. Und dadurch würden sie mehr gute Noten in der Schule bekommen.
- Außerdem haben einige heutzutage Eltern nicht so viel Zeit für ihre Kinder. Sie können ihren Kindern bei der Hausarbeit überhaupt nichts helfen. Mit

diesem Programm können Kinder Mathematik allein üben. Die Hilfe von den Eltern ist nicht mehr nötig, weil alle schriftlichen Rechenverfahren selbständig überprüft werden. Das Programm kennt die richtigen Regeln und weiß die richtige Antwort. Der Schüler kann selbständig arbeiten und hat mit dem Lernprogramm immer einen kompetenten Lernpartner zur Verfügung.

## 1.2. Lernaktivitäten

Es gibt heutzutage viele verschiedene Weise, um sich die Lernaktivitäten zu verteilen. Webseiten, Foren, Wikis, Chats, Mail, Blogs, Video Portal... sind die guten Werkzeuge für die Verteilung von Online-Lernaktivitäten. Allein in Deutschland wird die Zahl der Blogger auf ca. 250.000 geschätzt, die größte Online Community *MySpace* hat weltweit ca. 100 Millionen Mitglieder. Auf das Video Portal *Youtube* werden von den Nutzern täglich 70.000 neue Videos hochgeladen und viele davon sind die „*Tutorial-Videos*“ z.B. für die Fachbereiche Photoshop, 3D, Flash, etc....Die kollaborative Netz Enzyklopädie *Wikipedia* enthält inzwischen über 5 Millionen Artikel in über 200 Sprachen, davon mehr als 500.000 auf Deutsch <sup>1</sup>. Es gibt nicht wenige Mitglieder diesen oben erwähnten „*Social Webs*“, die Kinder unter 14 Jahre alt sind.

Bei den „*Social Webs*“ findet man meist die kostenlosen Produkte, und auf dem Markt für Lernsoftware, für die nicht kostenlosen Produkte, wie sieht es aus? Der Bundesverband Informationswirtschaft, Telekommunikation und neue Medien (BITCOM) teilte mit, dass der Umsatz mit Lernsoftware im Jahr 2007 in Deutschland um rund 16% auf 49 Millionen Euro gestiegen ist. Wenn im Jahr 2006 wurden die Lernsoftwares rund 42 Millionen Euro verkauft, war diese Anzahl im Jahr 2007 49 Millionen. Und die Anzahl von Kindern, die die Lernsoftwares benutzt haben, ist sehr groß. 42% der Benutzer waren die Kinder unter 14 Jahre alt <sup>2</sup>.

---

<sup>1</sup> Friedrich Hagedorn – Lernen im Netz: Überschätzt oder Aufbruch in neue Welten? S.1

<sup>2</sup> BITCOM - <http://www.checkpoint-elearning.de/article/4625.html>. Stand 15.4.2008

Aus den oben erwähnten Ergebnissen können wir sagen, dass Kinder sehr positiv an den Online- bzw. Offline- Lernaktivitäten teilnehmen.

### **1.3. Die Nutzung von Lernprogramm durch Kinder**

Eine Untersuchung von Medienpädagogischer Forschungsverbund Südwest gibt uns einen Blick über die Nutzung von Lernprogramm durch Kinder im Alter von 6 bis 13 Jahren in Deutschland:

Von den computererfahrenen Kinder im Alter von Sechs- bis 13-Jahren haben im Jahr 2006 43% mindestens einmal pro Woche ein Lernprogramm verwendet <sup>3</sup>. Nimmt man die seltenen Nutzer dazu, so haben ca. 81% der Computernutzer bereits mit Lernen am Computer Erfahrungen gesammelt <sup>4</sup>.

Die Beschäftigung mit Lernprogrammen macht den Kindern mehrheitlich Spaß. 14% der Nutzer beschäftigen sich sehr gerne mit Lernprogrammen, weitere 55% gerne mit diesen Anwendungen. Nur etwa einem Drittel nutzen Lernprogramme weniger bzw. überhaupt nicht gerne. Inhaltlich geht es bei den Programmen insbesondere um das Schulfach Mathematik (67%) <sup>5</sup>.

---

<sup>3</sup> Medienpädagogischer Forschungsverbund Südwest, KIM – Studie 2006, S.40

<sup>4</sup> Medienpädagogischer Forschungsverbund Südwest, KIM – Studie 2006, S.29

<sup>5</sup> Medienpädagogischer Forschungsverbund Südwest, KIM – Studie 2006, S.40

## 2. Beschreibung des Programms

Das Programm „Stark in Mathe 3“ ist mit der Software *Flash Professional 8* von *Macromedia* realisiert.

Im Lernprogramm „Stark in Mathe 3“ soll der Benutzer in unterhaltsamer Form an Mathematik in der 3.Klasse geübt werden. Das Lernprogramm vermittelt in fünf Lernbausteinen den gesamten Lernstoff der Mathematik des 3.Grundschuljahres. Der Inhalt deckt nahezu vollständig die Lehrpläne ab.

Es gibt hunderte von Übungen in jedem Lernbaustein. Jeder Lernbaustein wird nach dem Lernstoff eingeordnet und führt den Schüler schrittweise zum Lernerfolg. Alle schriftlichen Rechenverfahren werden selbständig überprüft. Jeder Lernbaustein endet mit einem Spiel, das unabhängig von den mathematischen Übungen spielbar ist. Wenn der Schüler genug geübt hat, kann er sich mit den Spielen entspannen.

### 2.1. Art des Programms

„Stark in Mathe 3“ ist eigentlich ein Übungsprogramm. Ein Übungsprogramm dient zum Einüben von bereits vorhandenem Vorwissen. Mit diesem Programm kann der Schüler die Mathematik noch einmal einüben, was er in der Schule gelernt hat. Das Programm gibt eine Abfolge von Aufgaben und Abfragen vor. Der Schüler soll die Aufgaben lösen und er wird ein Feedback bekommen, indem er den Überprüfens-Button anklickt. Das Feedback ist meist in der Form von Richtig-Falsch-Angaben.

## 2.2. Lektionen und Themen.

Das Programm „Stark in Mathe 3“ bietet fünf Hauptthemen der Mathematik des 3.Grundschuljahres.

<b>Themen</b>	<b>Kapitel</b>
<b>Zahlen bis 1000</b>	<ul style="list-style-type: none"> <li>- Hunderterzahlen</li> <li>- Hunderterzahlen vergleichen und zerlegen</li> <li>- Zahlen darstellen</li> <li>- Zahlen folgen</li> <li>- Zahlen vergleichen und ordnen</li> <li>- Vorgänger und Nachfolger</li> <li>- Nachbar-Zehner, Nachbar-Hunderter</li> <li>- und 1 Spiel (Zahlenraten)</li> </ul>
<b>Addieren &amp; Subtrahieren</b>	<ul style="list-style-type: none"> <li>- Hunderter addieren</li> <li>- Zehner addieren</li> <li>- Einer addieren</li> <li>- Hunderter subtrahieren</li> <li>- Zehner subtrahieren</li> <li>- Einer subtrahieren</li> <li>- Rechnen bis 1000</li> <li>- und 1 Spiel (Finde die Unterschiede).</li> </ul>
<b>Multiplizieren &amp; Dividieren</b>	<ul style="list-style-type: none"> <li>- Multiplizieren</li> <li>- Dividieren</li> <li>- Dividieren mit Rest</li> <li>- Schrittweise multiplizieren</li> <li>- Schrittweise dividieren</li> <li>- Rechnen bis 1000</li> <li>- und 1 Spiel (Wer wird Millionär).</li> </ul>
<b>Geometrie</b>	<ul style="list-style-type: none"> <li>- Spiegeln</li> <li>- Spiegelachsen</li> <li>- Würfel</li> </ul>

	<ul style="list-style-type: none"><li>- Flächen</li><li>- und 1 Spiel (Tangram).</li></ul>
<b>Umrechnen</b>	<ul style="list-style-type: none"><li>- Längen umrechnen und vergleichen</li><li>- Längen mit Komma schreiben</li><li>- Gewichte umrechnen und rechnen</li><li>- Sachaufgaben mit kg &amp; g</li><li>- Geldwerte umrechnen und vergleichen</li><li>- Geldwerte mit Komma schreiben</li><li>- Kalender/Wochen/Tage/Stunden/Minuten</li><li>- Zeitpunkte und Zeitspannen</li><li>- und 1 Spiel (Erdkunde).</li></ul>

### 2.3. Einsatzorte

Einer der größten Vorteile beim Einsatz von Lernsoftware liegt in der Unabhängigkeit vom Lernort und einer Lehrperson. Durch die Verbreitung des Lernmediums oder den Zugang über das Internet ist das Lernen überall möglich. Das Programm kann sowohl im privaten Haushalt als auch in Schulen eingesetzt werden. Außerdem kann es noch im Internet erstellt werden.

### 2.4. Hardware- und Softwarevoraussetzungen

Dieses Programm funktioniert auf fast jedem PC, MAC oder Linux Rechner. Auch alte Hardware wird keine Probleme bereiten, weil Flash Spiele relativ wenig Rechenstärke benötigen. Dieses Programm sollte auch noch auf mehrere Jahre alten Rechnern laufen. Für dieses Programm wird mindestens die folgende Hardwareanforderung empfohlen:

<b>Windows</b>	<b>Macintosh</b>	<b>Linux</b>
Bildschirmauflösung: 852 x 480	Bildschirmauflösung: 852 x 480	Bildschirmauflösung: 852 x 480
Prozessor mit mindestens 450 MHz	Prozessor mit 500 MHz oder schneller	Prozessor mit 800 MHz oder schneller
128 MB RAM	128 MB RAM	512 MB RAM, 128 MB Grafikspeicher
Soundkarte	Soundkarte	Soundkarte
CD-ROM	CD-ROM	CD-ROM

Wenn der Schüler eine CD vom Programm benutzt, muss er das Programm oder Flash Player nicht installieren. Das Programm wird automatisch starten, nach dem die CD in der CD-ROM eingelegt wurde.

Wenn der Schüler das Programm durch Internet benutzt, braucht er eine Internetverbindung Dual-ISDN oder besser z.B. LAN oder DSL. Das Programm kann auch mit geringeren Bandbreiten, also z.B. 56 kb Modem, betrachtet werden. Dieses Programm wird auf die gängigen Browser angepasst. Die optimale Darstellung und Funktion von diesem Programm wird mit dem Browser Internet Explorer 7 oder Firefox 2 erreicht. Das Programm soll auch gut mit den anderen Browsers wie z.B. Internet Explorer 6, Netscape 7.x, Opera 7.11 oder Safari 3.x funktionieren.

Für die Betrachtung von Flash basierten Animationen in einem Browser muss ein Plug-In<sup>6</sup> der Flash Player installiert werden. Dieser kann kostenlos von der Macromedia Website (<http://www.macromedia.com>) heruntergeladen werden.

---

<sup>6</sup> Plug-Ins sind Software-Module, die sich einfach in den Browser integrieren lassen. Sie erweitern den Browser um die Fähigkeiten, eine Vielzahl von interaktiven Multimedia-Anwendungen anzuzeigen.

### **3. Didaktik**

#### **3.1. Zielgruppe**

Die Zielgruppe für das Programm „Stark in Mathe 3“ sind Grundschüler in Deutschland, Klassenstufe 3, also für die Kinder im Alter von 8 bis 9 Jahren. Das Problem dieser Zielgruppe ist vor allen Dingen, dass sich Kinder in dem Alter noch in einer frühen Entwicklungsphase befinden, was bedeutet, dass sich Einstellungen, Wünsche und Erwartungen häufiger ändern als bei einer älteren Zielgruppe. Kinder ändern ihre Meinung relativ schnell, was eben noch ganz toll war, kann im nächsten Moment schon wieder völlig "uncool" sein. Außerdem verlieren Kinder ganz schnell ihre Konzentration. Wenn das Lernprogramm keine Aufmerksamkeit der Kinder erreichen und bewahren könnte, würden Kinder das Programm nicht mehr anfassen. Je nach Alter haben Kinder auch unterschiedliche Bedürfnisse und stellen verschiedene Ansprüche an Lernprogramm. Wenn Kinder meinen, dass Inhalte oder Design nicht zu ihrem Alter passen, werden sie das Programm ablehnen.

Das Programm ist hauptsächlich für das Lernen in der Schule oder für die private Nutzung gedacht. Mit diesem Programm können Kinder allein lernen, oder mit den Freunden in einer kleinen Gruppe zusammenlernen.

„Stark in Mathe 3“ ist sehr einfach zu benutzen. Um das Programm gut zu benutzen, braucht Nutzer nur eine wenige Kenntnisse im Umgang mit dem Computer. Mit 81% Kindern in Deutschland, die schon bereits mit Lernen am Computer Erfahrungen gesammelt haben, glaube ich, dass dieses Lernprogramm problemlos benutzt werden soll.

## 3.2. Lehrziele und Lehrinhalte

### 3.2.1. Lehrziele

Die Lehrziele für Lernprogramm, die sich nach Zielen des Schulunterrichts richten, können an den Lehrplänen der Bundesländer angelehnt werden. Um die Lehrziele des Lernprogramms „Stark in Mathe 3“ festzulegen, mussten zuerst die Lehrpläne des Faches Mathematik 3.Klasse näher beleuchtet werden.

Am Ende von Klasse 3 sollen möglichst alle Kinder die folgenden Fähigkeiten und Fertigkeiten erworben haben: <sup>7</sup>

#### **Im Zahlenraum bis 1000**

- Die Mächtigkeit von Mengen und Zahl abschätzen.
- Zahlen unterschiedlich darstellen.
- Ziffernfolgen erläutern.
- mit der Stellenwerttafel und am Zahlenstrahl arbeiten.
- Nachbarzahlen, Nachbarzehner und Nachbarhunderter finden.
- Zahleigenschaften und Zahlbeziehungen kennen.

#### **Rechnen**

- das Einspluseins gedächtnismäßig sicher beherrschen.
- Addition und Subtraktion mit zwei- und dreistelligen Zahlen im Kopf, halbschriftlich und schriftlich sicher ausführen.
- das Einmaleins und die dazugehörigen Divisionsaufgaben schnell und sicher lösen.
- eine Menge in gleichmächtige Teilmengen zerlegen und den Vorgang beschreiben.
- das Zehnereinsmaleins und die dazugehörigen Divisionsaufgaben schnell und sicher lösen.
- zweistellige Zahlen mit einstelligen Zahlen multiplizieren.
- zweistellige Zahlen durch einstellige Zahlen dividieren,

---

<sup>7</sup> Beispielaufgaben für Vergleichsarbeiten im Fach Mathematik in der Klasse 3 – Freie und Hansestadt Hamburg, Behörde für Bildung und Sport Amt für Bildung.

- Rechenvorteile und Zusammenhänge zwischen den Aufgaben nutzen (z.B. bei Tauschaufgaben, Nachbaraufgaben, Analogaufgaben, Umkehraufgaben, Aufgabenfamilien),
- Lösungswege beschreiben,
- Leerstellen bestimmen.
- beim Sachrechnen die richtigen Rechenoperationen ansetzen und durchführen,
- Aufgaben übersichtlich und lesbar aufschreiben.

### **Geometrie**

- Flächen und Körper zerlegen, auslegen, zusammensetzen,
- Flächen und Körper in Bezug auf Größe und Form beschreiben,
- Körpernetze herstellen und untersuchen,
- Lagebeziehungen erkennen und beschreiben,
- Symmetrien erkennen; die Lage von Spiegellinien erkennen,

### **Größen / Sachrechnen**

- mit Geldbeträgen – auch in Kommaschreibweise – Rechenoperationen durchführen,
- Preise und Gebühren des täglichen Lebens erkunden,
- die Längenmaße Kilometer, Meter, Zentimeter und die Gewichtsmaße Gramm und Kilogramm kennen und sinnvoll anwenden,
- mit Maßeinheiten – auch in Kommaschreibweise – Rechenoperationen durchführen,
- Zeiteinteilungen handhaben und beschreiben: Minuten, Stunden, Tage, Wochen, Monate, Jahr,
- Zeitspannen und Zeitpunkte berechnen,
- Zahlen und Zahlbeziehungen und Formen in der Umwelt entdecken und beschreiben,
- Pläne, Tabellen, Listen, Situationsskizzen „lesen“ und handhaben.

Bei der Festlegung der Lehrziele des Programms „Stark in Mathe 3“ wurden die oben genannten Lehrziele beachtet und ausgewogen. Fast alle Anforderungen im Mathematikunterricht von Klasse 3 wurden in diesem Programm erstellt.

### 3.2.2. Lehrinhalte und Struktur eines Inhaltsbereichs

Zur Festlegung der Lehrinhalte können wie bei den Lehrzielen auch die Lehrpläne der Bundesländer entnommen werden. Für dieses Programm wurden folgende Inhaltsbereiche ausgewählt:

- **Zahlen bis 1000**
- **Addieren und Subtrahieren**
- **Multiplizieren und Dividieren**
- **Geometrie**
- **Größen und Sachaufgaben.**

Zu jedem dieser Inhaltsbereiche wurden Aufgaben mit unterschiedlichem Anforderungsniveau, Schwierigkeitsgrad formuliert. Es gibt eine große Menge von Übungsaufgaben, die gelöst werden müssen. Bei mehreren anderen Lernprogrammen kann man nur zur nächsten Übung gelangen, wenn man die aktuelle Übung erfolgreich bewältigt hat. Beim Programm „Stark in Mathe 3“ ist es nicht so, die Aufgaben im Programm funktionieren unabhängig voneinander. Der Nutzer kann die Aufgaben auswählen, welche er lösen möchte und per nur einen Klick kann er wieder zum aktuellen Inhaltsbereich oder auch zum Startmenü zurückkommen. Der strukturelle Ablauf eines Inhaltsbereichs ist in Abbildung 1 „*Struktur eines Inhaltsbereichs*“ dargestellt.

### 3.2.3. Struktur einer Aufgabe

Alle Aufgaben beginnen ohne Unterschied. Das Programm gibt eine Aufgabe, die durch den Nutzer gelöst werden muss. Der Nutzer soll das Aufgabe lösen und sich das Ergebnis bestätigen. Nachdem Bestätigung wird er eine Rückmeldung bekommen. Die erste Art der Rückmeldung ist nur in der Form von Richtig-Falsch-Angaben, ohne Tipp. Die zweite Art der Rückmeldung ist auch in der Form von Richtig-Falsch-Angaben aber mit Tipps (diese Art finden wir in einigen Aufgaben des Lernbereichs „Zahlen bis 1000“ und „Geometrie“). Der Nutzer kann das Ergebnis zu jeder Zeit verändern und soll sich erneut bestätigen und dann wird er eine neue Rückmeldung bekommen. Der strukturelle Ablauf einer Aufgabe ist in Abbildung 2 „*Struktur einer Aufgabe*“ dargestellt.

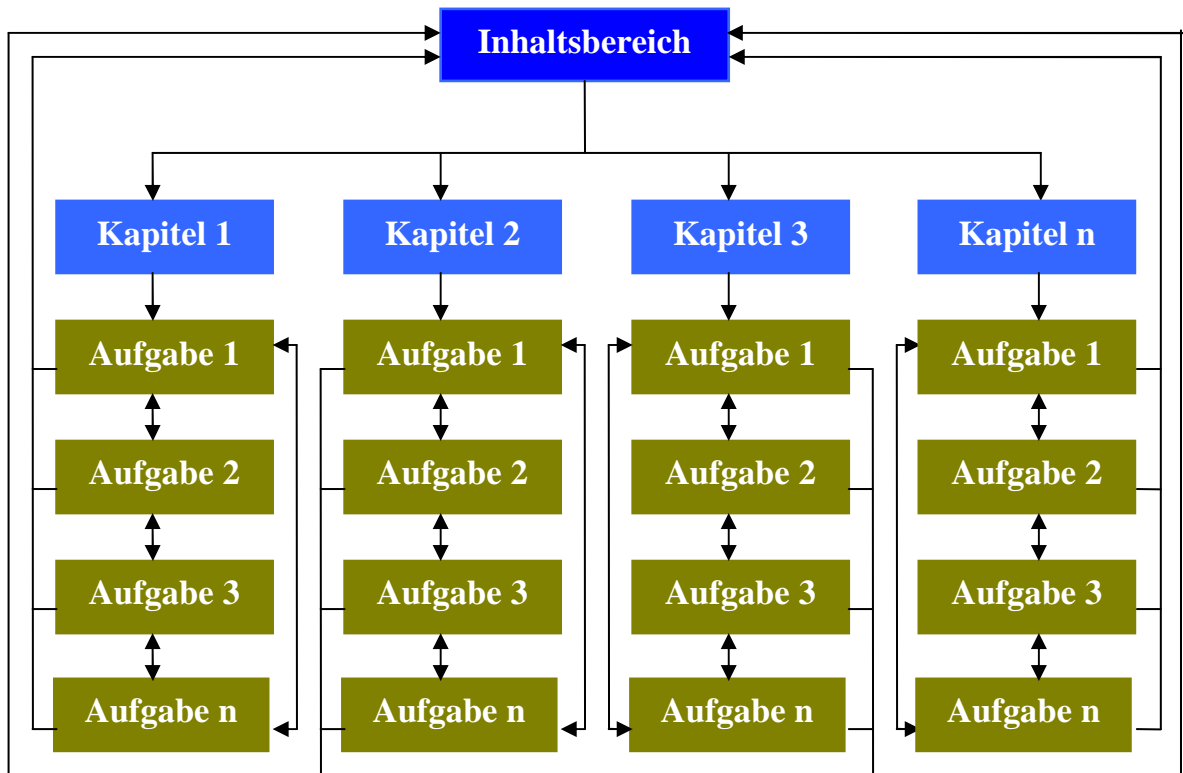


Abb. 1: Struktur eines Inhaltsbereichs.

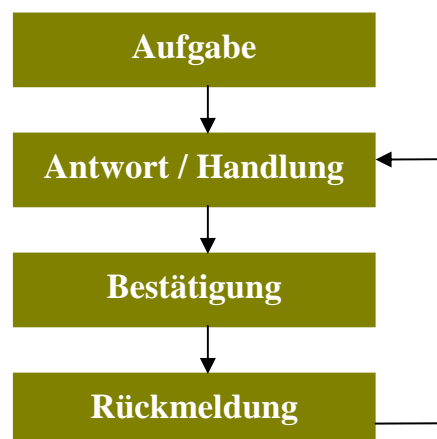


Abb. 2: Struktur einer Aufgabe

### 3.2.4. Aufgabentypen

Das Programm hat einige unterschiedliche Aufgabentypen, z.B. Lückentexte, Auswahlaufgaben (Multiple-Choice, mit einer oder mehreren richtigen Antworten), oder spielerische Lernformen. Aber die meisten Aufgaben sind Lückentext-Aufgaben.

Bei einer Lückentext-Aufgabe soll die Antwort in die Lücken der vorgegebenen Aufgabe eingegeben werden. Über den Überprüfens-Buttons kann der Schüler seine Lösung überprüfen. Bei diesem Aufgabentyp kann der Schüler seine Antwort zu jeder Zeit verändern. Die letzte Antwort wird nur nach der Bestätigung des Schülers festgelegt. Ein Beispiel des Typs Lückentext-Aufgabe ist in Abbildung 3 dargestellt.

Bei einer Multiple-Choice-Aufgabe soll der Schüler eine von mehreren vorgegebenen Antworten auf eine Frage auswählen, von diesen Antworten sind eine oder mehrere richtig. Bei diesem Aufgabentyp kann der Schüler seine Auswahl nicht verändern. Nach der Auswahl wird die Antwort sofort festgelegt. Ein Beispiel einer Multiple-Choice-Aufgabe ist in Abbildung 4 dargestellt.

Die Aufgaben im Programm sind an den Aufgaben in den Teil 1 und Teil 2 des Arbeitsheftes „Mathematik begreifen 3. Schuljahr“ von Ernst Klett Grundschulverlag angelehnt.

Macromedia Flash Player 8  
Datei Ansicht Steuerung Hilfe

☀️ 🐦

### Addieren & Subtrahieren

Hunderter addieren

1/3. Lege und rechne.

$200 + 400 = \square$	$348 + 300 = \square$
$100 + 800 = \square$	$219 + 500 = \square$
$500 + 500 = \square$	$527 + 200 = \square$
$400 + 300 = \square$	$664 + 100 = \square$
$300 + 500 = \square$	$475 + 400 = \square$

🐛

✓ ⏪ ⏩ ✗

Ende

Abb. 3: Eine Lückentext-Aufgabe

Macromedia Flash Player 8  
Datei Ansicht Steuerung Hilfe

☀️ 🐦

### Geometrie

Spiegelachsen

1/4. Klicke die Linien an, die Spiegelachsen sind.

🐛

✓ ⏪ ⏩ ✗

Ende

Abb. 4: Eine Multiple-Choice-Aufgabe

### 3.3. Lernmotivation

Bei der Erstellung von Lernprogramm ist die Lernmotivation des Benutzers sehr wichtig. Größere Motivation bedeutet einen besseren Lernerfolg. Im Allgemeinen differenziert man zwischen intrinsischer und extrinsischer Lernmotivation. Intrinsische Lernmotivation ergibt sich aus dem Material an sich. Ein Lernstoff wird gelernt, weil er interessant ist, weil Lernen Spaß macht. Anders ist es bei extrinsisch motivierten Lernern. Diese lernen, um bestimmte Ziele zu erreichen, wie z.B. das Erhalten des Arbeitsplatzes oder das Bestehen einer Prüfung...

Bei den extrinsisch motivierten Lernern ist das Lernen zuerst eine Pflicht. Diese Lerner finden meist keinen Spaß beim Lernen. Kinder gehören zu dieser Gruppe. Bei dieser Nutzergruppe muss das Lernprogramm versuchen, den Motivationsgrad durch eine ansprechende Gestaltung der Benutzeroberfläche, einer klaren Navigation und interessanten didaktischen Interaktionen zu steigern.

Eine Frage wurde gestellt: wie genau erreicht man eine Steigerung der Lernmotivation beim Benutzer? Dazu entwickelte John Keller in den 80er Jahren das ARCS-Modell. Das ARCS-Modell ist ein motivationales Instruktionsdesign fürs E-Learning <sup>8</sup>. In dem Modell beschreibt Keller die Merkmale, die eine motivierende Lernumgebung ausmachen. Das ARCS-Modell dient häufig als Grundlage für die motivationale Gestaltung von multimedialen Lernangeboten.

Keller beschreibt Motivation als zentrales Element des Instruktionsdesigns eines Lernangebots. Er definiert vier Motivationsfaktoren <sup>9</sup>:

- **A**ttention
- **R**elevance
- **C**onfidence
- **S**atisfaction

---

<sup>8</sup> Wikipedia, ARCS-Modell, <http://de.wikipedia.org/wiki/ARCS-Modell>. Stand 15.03.2008.

<sup>9</sup> Mehr: <http://www.tu-chemnitz.de/phil/elearning/studentenprojekte/motivation/arcs.htm>. Stand 15.03.2008.

Zuerst soll das Lernprogramm die Aufmerksamkeit des Benutzers erlangen (Attention). Der Inhalt soll für den Nutzer interessant aufbereitet werden und natürlich auch relevant sein (Relevance). Nun soll das Lernprogramm die Erfolgsszuversicht (Confidence) unterstützen und die Zufriedenheit des Nutzers (Satisfaction) herstellen.

Wie im Kapitel 3.1. „Zielgruppe“ beschrieben, handelt es sich bei der Zielgruppe des Lernprogramms „Stark in Mathe 3“ um Kinder, welche sehr viele Motivationen beim Lernen brauchen. Kinder sind die extrinsisch motivierten Lerner. Sie besitzen eine relativ geringe Lernmotivation aber auch eine geringe Konzentrationsfähigkeit. Deshalb ist es nötig, besonderen Wert auf die Motivation der Kinder zu legen. Im weiteren Verlauf wird beschrieben, über welche Motivationsfaktoren die Kinder im Lernprogramm motiviert werden. Die benutzten Motivationsfaktoren werden auch hier dem ARCS-Modell zugeordnet.

### **Attention**

In einem Lernprogramm ist es wichtig, die Aufmerksamkeit des Lerners zu erringen und zu bewahren. Um beim Lerner Aufmerksamkeit zu erringen, muss man seine Neugier wecken. Hier ist es möglich, anhand eines ansprechenden Oberflächendesigns, durch Multimedia (z.B. Ton oder Animationen) oder auch durch Interaktionen, das Interesse des Nutzers zu erlangen. Bei einem Programm für Kinder sind Animationen und Soundeffekten besonders wichtig. Der Einsatz von Animationen und Soundeffekten auswirkt sich positiv auf das Interesse der Kinder. Sie geben einen guten ersten Eindruck und ermutigen die Kinder, sich mit dem Programm zu beschäftigen.

Das Programm „Stark in Mathe 3“ startet mit den verbindenden Animationen und Soundeffekten den unterschiedlichen Elementen, wie z.B. die Hügel, die Sonne ... und die Bleistifte. Die Bleistifte sind die Buttons, die die Lernbereiche präsentieren. Jeder Bleistift wird auch animiert, um die Aufmerksamkeit des Benutzers zu erringen. Wenn der Benutzer einen Bleistift anklickt, wird ein

Lernbereich interessant und unerwartet durch eine animierte Tafel präsentiert. Diese solche Sache macht Kinder Spaß.

Wie man an Spielzeug, Kindersendungen oder Internetseiten für Kinder erkennen kann, lieben sie bunte Sachen. Gesättigte Farben und viele Farbkontraste werden bei der Gestaltung eingesetzt. Außerdem sind Comics bei Kindern sehr beliebt. Mathematik ist normalerweise ein trockenes Fach. Um „Stark in Mathe 3“ nicht als ein zu ernstes und trockenes Programm wirken zu lassen, wurden diese beiden Elemente in diesem Programm miteinander kombiniert. Die Sonne, die Bleistifte, die Tafel, die Maus, der Vogel...leben in einer sehr bunten Comicwelt, die die Nutzer im ganzen Lernverlauf sehen können.

Die animierte Maus in diesem Programm, die sich an der unteren Ecke der Tafel bewegt, spielt nur eine Rolle, das Interesse der Kinder am Programm zu gewinnen. Ich glaube, dass diese animierte Maus sich sehr positiv auf das Interesse der Kinder auswirkt.



*Abb. 5: Die Maus*

### **Relevance**

Wie bei der Zielgruppenanalyse im Kapitel 3.1. „Zielgruppe“ erwähnt, werden Kinder das Programm ablehnen, wenn Inhalte oder Design nicht zu ihrem Alter angepasst sind. Deswegen muss man auf das Alter und den Bildungsgrad der Zielgruppe besonders achten. Natürlich können auch die älteren Kinder die Lernsoftware zur Festigung oder zur Wiederholung nutzen, aber das bunte, verspielte Design zeigt, dass die Lernsoftware den Kindern im Alter von 8 bis 9 Jahren entspricht. Die Lehrinhalte sind natürlich an die Kenntnisse der Schüler angepasst, weil die Aufgaben im Programm an den Lehrplänen des Faches Mathematik 3.Klasse der Bundesländer angelehnt sind. Bei einigen schwierigen Aufgaben oder bei den Aufgaben, die die unklaren Forderungen haben, bekommt der Schüler am Anfang der Aufgabe immer ein Beispiel. Dieses Beispiel macht die Aufgabe leichter zu verstehen.

Die Spiele dürfen in einem Programm für Kinder nicht fehlen. Im Programm „Stark in Mathe 3“ beendet jeder Lernbaustein mit einem Spiel, das unabhängig von den mathematischen Übungen spielbar ist. Wenn der Schüler genug geübt hat, kann er sich mit den Spielen entspannen. Zwei Spiele von den passen ganz gut zum Lernthema. Zum Beispiel das Spiel „Zahlenraten“ im Lernthema „Zahlen bis 1000“. Mit diesem Spiel kann der Schüler die Aufgaben „Zahlen vergleichen“ und „Zahlen folgen“ festigen. Das Spiel „Tangram“ im Lernbaustein „Geometrie“ ist ein Formspiel. Mit diesem Spiel spielt der Schüler noch mal mit den geometrischen Figuren, die er im Unterricht kennengelernt hat. Die anderen Spiele sind die entsprechenden Spiele zu dem Alter der Schüler in der 3.Klasse.

### **Confidence**

Um das Vertrauen der Schüler zu gewinnen, muss besonders auf ein klares und übersichtliches Design und auf eine konsistente Navigation gelegt werden. Deshalb habe ich versucht, intuitive Interaktions- und Navigationselemente und eine übersichtliche Navigation in der Lernsoftware „Stark in Mathe 3“ zu gestalten. Zum Beispiel haben die Tasten der Steuerung, die an der Tafel stehen, immer dieselbe Funktion und Position. Der Nutzer muss sich nur einmal mit der Steuerung vertraut machen. Die Struktur des Lernangebots wurde klar dargeboten. Die Aufgaben, Texte, Rückmeldungen wurden auf einen absoluten ruhigen Hintergrund mit sehr gut lesbarer Schrift präsentiert. Die Eingabetexte, die von dem Nutzer eingetippt werden, werden mit einer anderen Farbe (gelbe Farbe) eingesetzt, damit der Nutzer erkennen kann, was er eingetippt hat. Nach der Antwortbestätigung wird die Antwort durch sofortige Rückmeldung evaluiert.

Außerdem wird der Nutzer bei jedem Kapitel mitgeteilt, wie viele Aufgaben ihn erwarten. Der Nutzer kann selbst entscheiden, welchen Teil des Lehrstoffs er bearbeiten möchte und er kann das gerade bearbeitete Kapitel bzw. das Lernprogramm zu jeder Zeit abbrechen. So kann der Nutzer seinen Lernverlauf selbst kontrollieren.

**Satisfaction**

Der vierte und letzte Motivationsfaktor ist die Herstellung von Zufriedenheit (Satisfaction). Für die Kinder zählt weniger der Lernerfolg bei Benutzung einer Lernsoftware. Auf diesen legen eher die Lehrer und Eltern großen Wert. Kinder wollen, dass der Umgang mit der Lernsoftware in erster Linie Spaß macht. Um dies zu erreichen, können bei guten Leistungen kleine Belohnungen wie ein Lob oder Animationen gegeben werden. Diese Belohnungen motivieren den Schüler und machen ihn glücklich mit seinen Lernleistungen. Im Programm „Stark in Mathe 3“ bekommt der Schüler Rückmeldungen nach der Bestätigung seiner Antwort. Diese Rückmeldung ist eine Kombination von mehreren Elementen: die Gesichter, Sound und der animierte Vogel. Wenn der Schüler alle Aufgaben richtig gelöst hätte, erscheinen die grünen lachenden Gesichter und Sound. Gleichzeitig springt der Vogel auf und er sieht richtig glücklich aus. Er erfreut sich mit der Leistung des Schülers. Für die meisten Kinder ist das Glück des Vogels eine schöne Belohnung.

## 4. Gestaltung

Wenn das didaktische Konzept feststeht, ist es notwendig, sich über die visuelle Gestaltung der Anwendung Gedanken zu machen. Die Gestaltung der Lernanwendung hat eine Funktion, die Verbindung zwischen Lehrinhalten und Lernenden zu schaffen. Damit der Lernende in seinem Lernprozess bestmöglich unterstützt wird, ist eine ansprechende Gestaltung besonders wichtig.

### 4.1. Screendesign

#### 4.1.1. Startseite

Wie bereits im Kapitel 3.3. „Lernmotivation“ beschrieben, startet das Programm „Stark in Mathe 3“ mit den verbindenden Animationen und Soundeffekten den unterschiedlichen Elementen, wie z.B. die Hügel, die Sonne, der Regenbogen... und die Bleistifte. Hier werden dem Schüler die Lehrinhalte durch die animierten Bleistifte präsentiert. Comics erfreuen sich bei Kindern sehr stark. Sie sind einfach zu verstehen, haben wenig Text und sind bunt. Die Startseite dieses Programms wurde auch mit einer bunten Comicwelt gestaltet. Für den Name des Programms wird die Schriftart *VNI-Whimsy* eingesetzt. Diese Schriftart ist sehr comichaft und wurde in verschiedenen Farben gemacht. So haben die Kinder bei der Benutzung des Programms das Gefühl, etwas Vertrautes vor sich zu haben. Das nachfolgende Bild zeigt die Startseite des Programms „Stark in Mathe 3“.

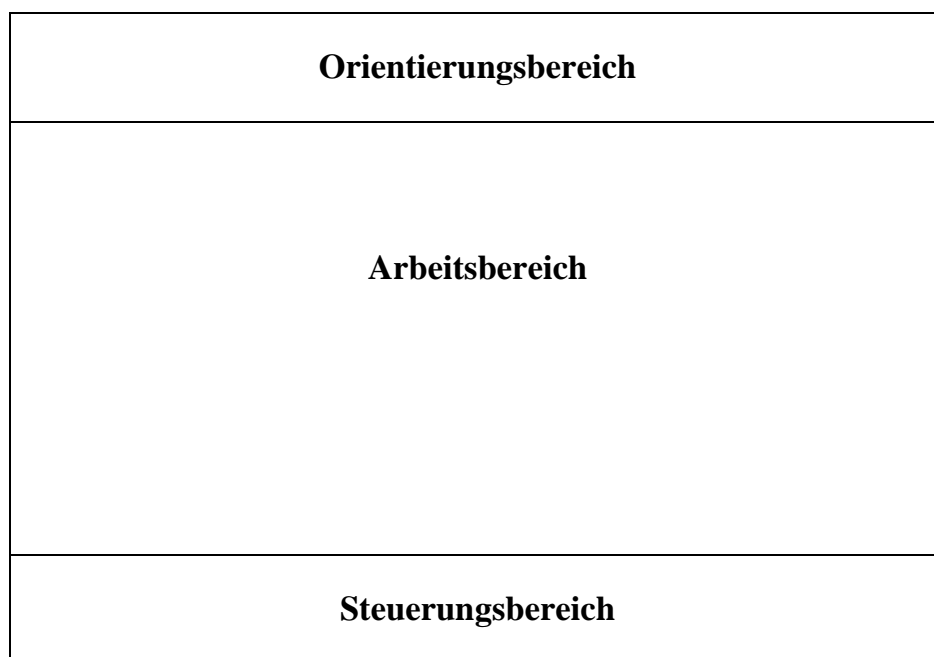


Abb. 6: Startseite des Programms „Stark in Mathe 3“

### 4.1.2. Arbeitsseite

Für die Gestaltung einer Lernoberfläche ist die Aufteilung des Bildschirms in Funktionsbereiche wichtig. Die Aufteilung des Bildschirms bestimmt die Position der Elemente des Programms. Wichtig beim Aufbau von Arbeitsseiten ist die klare Strukturierung in die folgenden drei Bereiche:

- Orientierungsbereich
- Arbeitsbereich
- Steuerungsbereich



*Abb. 7: Bildschirmaufteilung eines Lernprogramms*

Im Orientierungsbereich kann der Nutzer ablesen, an welcher Stelle im Programm er sich gerade befindet bzw. welches Thema gerade gearbeitet wird. Dies ist meist ein kleinerer Bereich oder eine Überschrift am Bildschirmrand.

Der Arbeitsbereich nimmt den größten Teil des Bildschirms ein und ist der Bereich, in dem die Lerninhalte präsentiert werden. In diesem Bereich werden Informationen bzw. Aufgaben dargestellt.

Der Steuerungsbereich dient dazu, sich im Programm zu bewegen. Dieser Bereich enthält Navigations- und Steuerelemente, hier sollten die globalen Funktionen des Programms angezeigt werden.

Die nachfolgende Abbildung zeigt die Aufteilung der Funktionsbereiche der Tafel im Programm „Stark in Mathe 3“.



Abb. 8: Bildschirmaufteilung des Programms „Stark in Mathe 3“

Die Funktionsbereiche des Programms wurden deutlich aufgeteilt und im ganzen Lernverlauf nicht verändert. Die Tafel ist für die Schüler eine sehr vertraute Sache. Und die Aufteilung der Funktionsbereiche der Tafel in diesem Programm ist auch sehr ähnlich wie die Aufteilung der Tafel, die die Schüler täglich in der Schule sehen.

In diesem Programm wurden die Text, Grafik, Animation und Ton eingesetzt. Hier wurde es darauf geachtet, dass ein starker Kontrast zwischen den Farben von Text und Hintergrund besteht, um einen ungestörten Lesefluss zu garantieren. Für den Inhalt des Programms wurde die Schriftart *Comic Sans MS* mit einer Schriftgröße ab 18 pt in Weiß eingesetzt. Sie ist gut lesbar und für ein Lernprogramm für Kinder gut geeignet. Für Hintergrund wurde ein dunkler Farbton (dunkelblau) benutzt, um einen Kontrast zu den übrigen Teilen des Programms deutlich zu machen.

Die Grafiken findet man unter zwei Spielen des Programms („Finde die Unterschiede“ und „Erdkunde Spiel“). Beim Einsatz von Karten im Spiel

„Erdkunde Spiel“ fiel die Wahl auf das Vektorgrafikformat Shockware – Flash. Dieses Format ist sehr komprimiert und daher sehr klein.

Ton wurde beim Beginn des Programms benutzt. Anfangs wird dieser bei den Animationen eingesetzt. Dann gelangt der Nutzer in die Lernthemen. Auch hier gibt es einige Soundeffekte zur Motivation. In den Aufgaben wird Ton nur noch zur Rückmeldung der Benutzerantwort eingesetzt. Eine Hintergrundmusik würde in einem Lernprogramm ablenken, deshalb ist sie in diesem Programm nicht nötig. Es gibt verschiedene Sounds für richtige oder falsche Antworten.

Die Animationen in dieser Seite findet man in einigen Aufgaben beim Lernthema „Geometrie“ und bei der Rückmeldung (der Vogel). Das Aussehen und die Bewegung des Vogels zeigen dem Schüler, dass er die Aufgabe richtig oder falsch gelöst hat. Die nachfolgende Abbildung zeigt die drei Gefühlszustände des Vogels:

- Normal: warte auf die Antwort des Schülers.
- Traurig: wenn der Schüler eine Aufgabe falsch gelöst hat.
- Glücklich: wenn der Schüler alle Aufgaben richtig gelöst hat.



*Abb. 9: Die Gefühlszustände des Vogels*

Außerdem wird die Rückmeldung noch durch die roten oder grünen Gesichter dargestellt. Das rote traurige Gesicht stellt eine falsche Antwort dar und im Gegenteil stellt das grüne lachende Gesicht eine richtige Antwort dar.



*Abb. 10: die gefühlsmäßigen Gesichter*

## 4.2. Navigation

Mit der Hilfe der Navigationsfunktionen kann der Nutzer auf die Lerninhalte zugreifen und auch alle anderen Programmfunktionen aufrufen. Ein unerlässlicher Bestandteil jeder Lernanwendung ist ein Hauptmenü, das dem Lernenden erst das Navigieren und Orientieren ermöglicht. Dieses Hauptmenü sollte von jeder Stelle im Programm zu erreichen sein. Im Programm „Stark in Mathe 3“ präsentiert sich das Hauptmenü als die animierten Bleistifte. Es gestattet den direkten Zugriff auf die Lernthemen und gibt dem Nutzer eine Übersicht über den verfügbaren Inhalt. Weil die Zielgruppe für das Programm die Kinder im Alter von 8 bis 9 Jahren ist, wurde das Hauptmenü in einer sehr bunten Comicwelt gestaltet. Jeder Bleistift hat einen eigenen Charakter und eine eigene Farbe. Sie sehen spaßig aus und geben einen guten ersten Eindruck und ermutigen die Kinder, sich mit dem Programm zu beschäftigen. Die nachfolgende Abbildung zeigt die Bleistifte aus dem Programm „Stark in Mathe 3“.

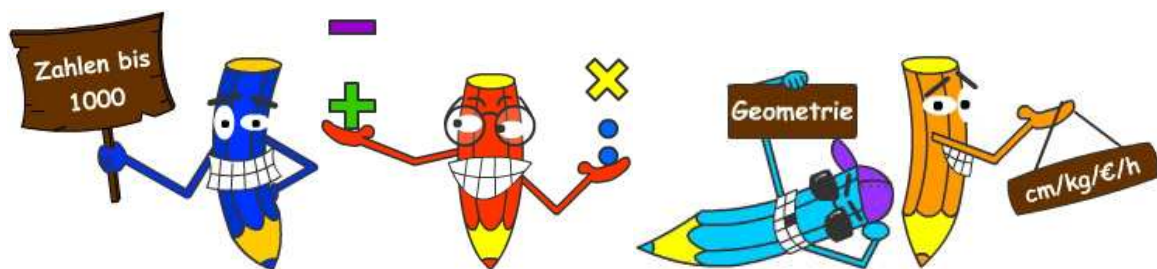


Abb. 11: die Bleistifte aus der Software „Stark in Mathe 3“

Bei Kindern ist das abstrakte Denken noch nicht so stark ausgeprägt. Deshalb sollte die Navigation so intuitiv wie möglich gehalten werden und die Navigationselemente einen starken Bezug zur Realität haben. Hier werden die Lerninhalte so einfach wie möglich gezeigt: z.B. der rote Bleistift wirft die Symbole von Addition, Subtraktion, Multiplikation und Division hoch und die anderen Bleistifte nehmen die Schilder, auf den werden die Lerninhalte geschrieben. Außerdem sind die Interaktionen z.B. durch Soundeffekte, Mauszeigeränderungen und Animationen gekennzeichnet.

Nach der Auswahl eines Lernthemas wird der Schüler ein weiteres Menü bekommen. Dieses Menü ist eine Liste von den Kapiteln. Dieses Menü wurde sehr

einfach gestaltet. Es gibt hier nur Texte. Die anklickbaren Elemente der Liste werden durch eine Mauszeigeränderung und durch die Veränderungen der Größe und der Farbe des Textes gekennzeichnet.

Nach der Auswahl eines Kapitels wird der Schüler an die Aufgaben gelangen. Hier gibt es die Menüleiste im Steuerungsbereich. Das grafische Design und die Art der Navigation wurden so intuitiv wie möglich gestaltet. Die Buttons haben überall die gleiche Funktion. Die Buttons sind von Anfang in Weiß, sie werden sich verfärben, wenn der Nutzer sich mit der Maus darüber bewegt. Die folgende Tabelle zeigt die Buttons in der Menüleiste des Programms, deren Assoziation, Bedeutung und Funktion in diesem Programm.

<b>Button</b>	<b>Assoziation</b>	<b>Bedeutung</b>	<b>Funktion</b>
	<i>Form:</i> Durchgestrichen = Ende, Schließen.  <i>Farbe:</i> Rot = Stopp, zurück.	Zurückkehren, Ende.	Mit diesem Button kommt der Nutzer auf das Menü von den Kapiteln zurück.
	<i>Form:</i> Pfeil nach rechts = weiter, nächster Schritt.  <i>Farbe:</i> Blau = Klarheit, losgehen.	Einen Schritt weiter, nächster Bildschirminhalt.	Mit diesem Button blättert der Nutzer zur nächsten Bildschirmseite auf dem gleichen Kapitel. Hier bekommt er die neuen Aufgaben.
	<i>Form:</i> Pfeil nach links = zurück, letzter Schritt.	Einen Schritt zurück, letzter Bildschirminhalt.	Mit diesem Button geht der Nutzer zur letzten Bildschirmseite auf dem

	<p><i>Farbe:</i> Orange = Achtung</p>		gleichen Kapitel zurück.
	<p><i>Form:</i> abgehakt = fertig, richtig, Bestätigung.</p> <p><i>Farbe:</i> Grün = Hoffnung, richtig, weiter.</p>	Fertig, Antwortbestätigung	Mit diesem Button bestätigt der Nutzer seine Antwort. Dann bekommt er die Rückmeldung.
	<p><i>Form:</i> ovaler Button mit Text „Ende“ steht drauf = Ende.</p> <p><i>Farbe:</i> Rot = Stopp, zurück.</p>	Ende, Zurückkehren.	Mit diesem Button beendet der Nutzer seine Arbeit und kommt zur Startseite zurück.

## 5. Technische Entwicklung

In diesem Kapitel wird die technische Entwicklung des Programms „Stark in Mathe 3“ beschrieben. Bei der technischen Entwicklung eines Programms muss zuerst geklärt werden mit welchem Werkzeug dieses erstellt werden soll. Dieses Programm wurde mit *Flash Professional 8* fertig gebaut. Einige Grafiken im Programm wurden mit *Adobe Photoshop CS2* bearbeitet, aber das wichtigste Werkzeug für dieses Programm ist *Flash Professional 8* von *Macromedia*. Deshalb werden im Folgenden das Programm *Macromedia Flash* und die technische Umgebung beschrieben.

### 5.1. Macromedia Flash

*Flash* von der Firma *Macromedia* ist ein leistungsstarkes Software-Werkzeug zur Entwicklung von Webanwendungen. Ursprünglich wurde es vor allem für multimediale Präsentationen im Web (z.B. Online-Werbeauftritte, Infos etc.) verwendet. Mit den zahlreichen Erweiterungen, besonders auch der integrierten Scriptsprache *ActionScript*, in den aktuellsten Versionen wurde *Flash* auch für interaktive Anwendungen und damit besonders auch E-Learning immer interessanter.

*Flash* ist zeitachsenorientiert aufgebaut und arbeitet mit der Metapher Film. Der Film wird über eine Zeitleiste realisiert. Bei der Medienentwicklung mit *Flash* werden vorproduzierte Medienobjekte (Text, Sound, Grafik, MovieClips etc.) in einer Bibliothek abgelegt und per *Drag-and-Drop* auf der so genannten *Bühne*, die die Arbeitsfläche darstellt, platziert. Das Verhalten der Objekte wird durch den Ablauf der Zeitleiste und durch den Objekten oder Zeitpunkten zugeordneten *ActionScript* Code bestimmt. Die Gestaltung und der Aufbau der verschiedenen Medienobjekte sowie die Programmierung von einfacherem Verhalten können grundsätzlich visuell durchführen. Anspruchsvollere, interaktive Anwendungen können mit *ActionScript* programmiert werden.

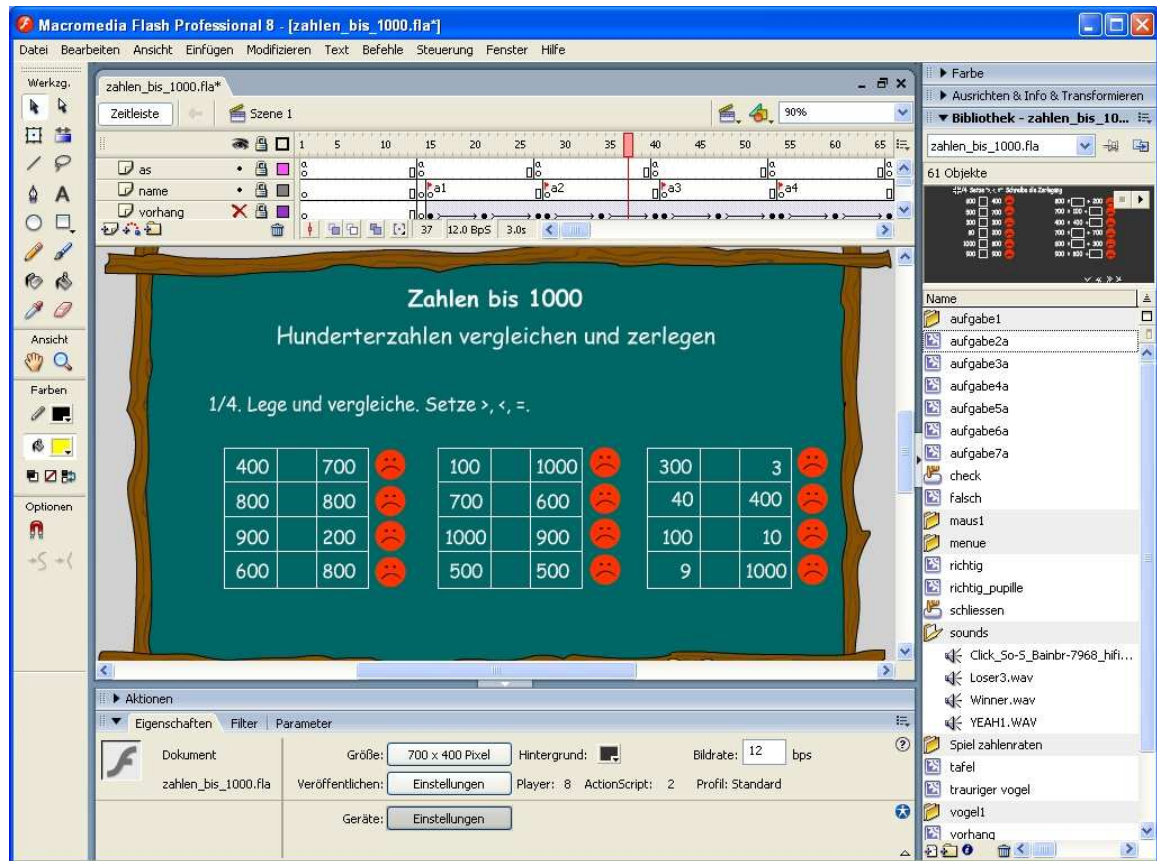


Abb. 12: Screenshot der Flash-Entwicklungsumgebung

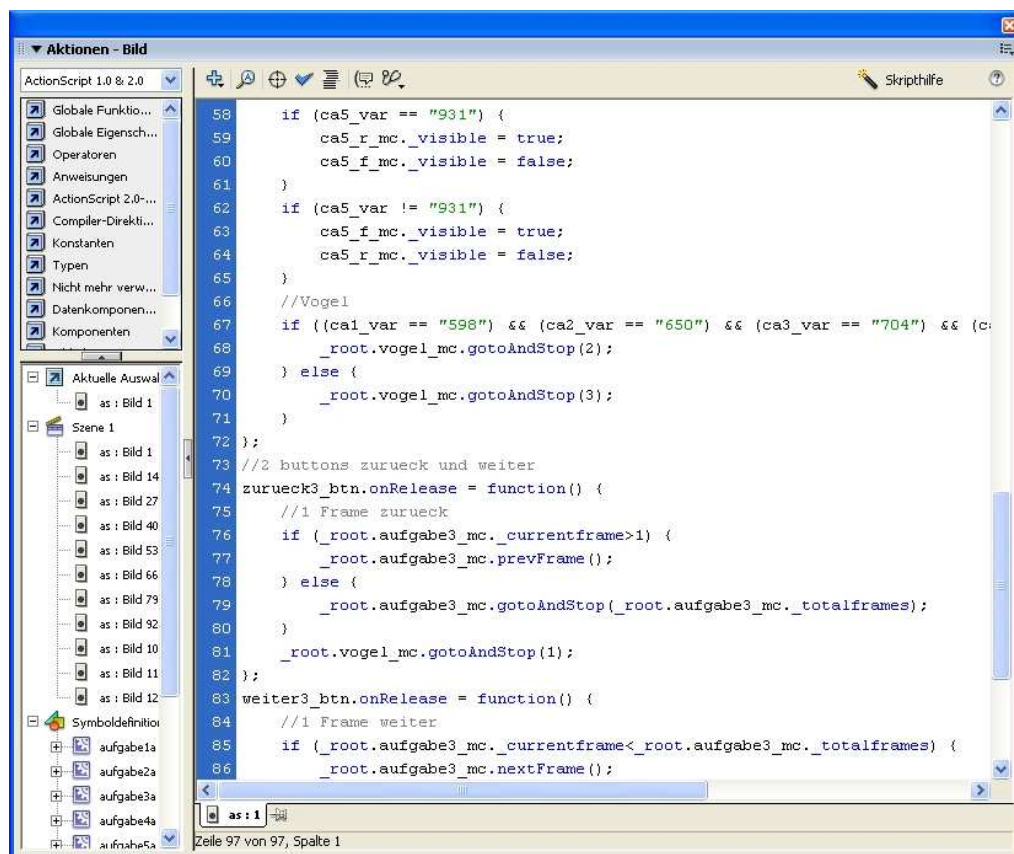


Abb. 13: Beispiel für ActionScript-Code

Die in der Entwicklungsumgebung erstellten Projekte lassen sich als Dokumente mit der Dateierweiterung *FLA* abspeichern. Sie enthalten lediglich die Quelldaten in Form von Objekten und ActionScript-Anweisungen, repräsentieren aber noch nicht den eigentlichen Flash-Film. Um die externe Wiedergabe zu ermöglichen, müssen die Datei zunächst in ein anderes Format kompiliert werden. Sie müssen in das *Shockwave Flash Format* exportiert werden (Filme im *Shockwave Flash Format* besitzen die Dateierweiterung *SWF*). Das Format stellt einen offenen Standard dar und erfährt als solcher mittlerweile auch Unterstützung durch andere Anwendungen.

Für die Wiedergabe der SWF-Dateien wird eine spezielle Software benötigt, der heißt Shockwave Flash Player. Dieser existiert einerseits in Form von Browser-Erweiterungen (PlugIns) und ermöglicht die Darstellung von Flash-Inhalten im Netz, andererseits als eigenständige Applikation<sup>10</sup>.

### **Möglichkeiten mit Flash:**

Mit den vielfältigen Funktionen in Flash können viele Anwendungsarten erstellt werden. Nachfolgen einige Beispiele:

- **Animationen:** Dazu gehören Werbeeinblendungen, Online-Grußkarten, Cartoons usw. Viele andere Arten von Flash-Anwendungen beinhalten auch Animationselemente.
- **Spiele:** Viele Spiele werden mit Flash erstellt. Bei Spielen werden in der Regel die Animationsfunktionen von Flash mit den Codefunktionen von ActionScript kombiniert.
- **Benutzeroberflächen:** Viele Website-Designer entwerfen Benutzeroberflächen mit Flash. Dazu zählen einfache Navigationsleisten, aber auch wesentlich komplexere Oberflächen.

---

<sup>10</sup> Mehr über Flash: [http://de.wikipedia.org/wiki/Adobe\\_Flash](http://de.wikipedia.org/wiki/Adobe_Flash)

- **Rich-Internet-Anwendungen:** Diese Anwendungen bieten eine funktionsreiche Benutzeroberfläche, mit der entfernt gespeicherte Daten über das Internet angezeigt und bearbeitet werden können. Hierbei kann es sich um Kalenderanwendungen, Preissucheanwendungen, Einkaufskataloge, Schulungs- und Prüfungsanwendungen oder andere Anwendungen handeln, die Remote-Daten in einer optisch ansprechenden Oberfläche anzeigen.
- **Lernanwendungen:** Es gibt immer mehr Lernanwendungen auf dem Markt, die mit Flash erstellt werden. Mit der Möglichkeit, die Medienelemente wie Bilder, Videos, Sounds, Texte usw. erstellt und importiert werden zu können, und mit der Hilfe von ActionScript, werden die Flash-Lernanwendungen immer populärer.

## 5.2. Programmierung

Im Rahmen der Entwicklung des Programms „Stark in Mathe 3“ kam es zu umfangreichen Programmieraufgaben. In nächsten Schritt werden einige besondere programmiertechnische Umsetzungen erläutert.

### 5.2.1. Animationen

Im Programm „Stark in Mathe 3“ gibt es mehrere Animationen, z.B. die Bewegungen von den Bleistiften, die Tafel, die Maus, der Vogel usw.... Diese Bewegungen sind die so genannte Bewegungs-Tweening. Sie ist hauptsächlich für Bewegungen zwischen Punkt A und Punkt B gedacht. Bei diesem Animationstyp muss man ein Start- und ein Endbild erstellen und überlassen es Flash, die entsprechenden Zwischenbilder zu berechnen. **Beispiel:** die Bewegung der Maus

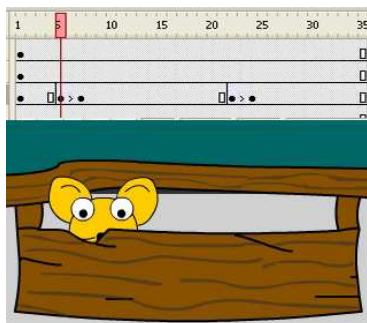


Abb. 14: Startbild der Bewegung

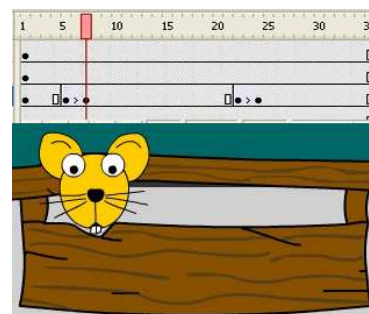


Abb. 15: Endbild der Bewegung

### 5.2.2. Loadmovie

Jedes Lernthema in diesem Programm wurde in einer einzelnen Datei erstellt. Diese Dateien wurden als externe SWF-Dateien gespeichert. Wenn ein Lernthema beim Nutzer ausgewählt wird, wird die externe SWF-Datei in den Hauptfilm geladen. Der Hauptfilm dieses Programms ist die Startseite und die externen Dateien werden auf die Tafel des Hauptfilms geladen. Um eine externe Datei zu laden, bietet Flash die *loadMovie*-Funktion. *LoadMovie*-Funktion lädt eine SWF-, JPEG-, GIF-, oder PNG-Datei in einen MovieClip in Flash Player, während die ursprüngliche SWF-Datei abgespielt wird.

Das folgende Quellecode-Beispiel zeigt einen *loadMovie*-Befehl. In diesem Befehl ist „bleistift1\_mc“ der Instanzname vom Button „Zahlen bis 1000“, „zahlen\_bis\_1000.swf“ ist der Name der externen SWF-Datei, und „load\_mc“ ist der Instanzname des MovieClips im Hauptfilm, wo die externe SWF-Datei geladen werden soll.

```
bleistift1_mc.onRelease = function() {  
    gotoAndPlay(201);  
    loadMovie("zahlen_bis_1000.swf", _root.load_mc);  
};
```

### 5.2.3. Lückentexte

Die meisten Aufgaben in diesem Programm sind Lückentext-Aufgabe. Der Schüler soll seine Antwort in die Lücken der Aufgabe eingeben, über den Überprüfens-Button kann er seine Lösung überprüfen. Um diese Aufgabe zu programmieren, braucht man zuerst einen Lückentext. Einem Flash-Dokument kann man verschiedene Arten von Text hinzufügen: statischen Text, dynamischen Text oder Eingabetext. Für den Aufgabebetyp wie Lückentext-Aufgabe soll man Eingabetext verwenden und dem Eingabetext einen festen Wert geben. Dieser feste Wert ist die richtige Lösung der Aufgabe. Wenn der Schüler eine Antwort eingeben würde, die anders als der feste Wert des Eingabetextes wäre, wird er eine Falsch-

Rückmeldung bekommen. Im Gegenteil wird er eine Richtig-Rückmeldung bekommen, wenn seine Antwort gleich dem festen Wert wäre.

**Beispiel:** Aus der Aufgabe „1/3. Lege und rechne“ beim Lernthema „Addieren & Subtrahieren“, Kapitel „Hunderter addieren“ zeigt das folgende Quellcode-Beispiel einen *ActionScript*-Code einer Lückentext-Aufgabe.

Die Aufgabe lautet: Lege und rechne.

200 + 400 =

```
check_bla_btn.onRelease = function() {  
    if (bla1_var == "600") {  
        bla1_r_mc._visible = true;  
        bla1_f_mc._visible = false;  
    } else {  
        bla1_f_mc._visible = true;  
        bla1_r_mc._visible = false;  
    }  
}
```

**Anmerkung:**

- check\_bla\_btn: der Instanzname des Überprüfens-Buttons.
- bla1\_var: der Variable-Name des Eingabetextes.
- bla1\_r\_mc: der Instanzname des grünen Gesichtes (Richtig).
- bla1\_f\_mc: der Instanzname des roten Gesichtes (Falsch).

Bei dieser Aufgabe wurde der Eingabetext einen Wert „600“ gegeben. Wenn der Eingabetext einen Wert gleich 600 bekommt, wird das grüne Gesicht sichtbar und das rote Gesicht unsichtbar. Wenn der Eingabetext einen Wert ungleich 600 bekommt, wird das rote Gesicht sichtbar und das grüne Gesicht wird unsichtbar.

#### 5.2.4. Multiple-Choice

Die Aufgaben vom Typ „Multiple-Choice“ kann man in mehreren Aufgaben beim Lernthema „Geometrie“ finden. Bei der Multiple-Choice-Aufgabe wählt der

Schüler eine von mehreren vorgegebenen Antworten auf eine Frage aus, von diesen Antworten sind eine oder mehrere richtig. Für die Programmierung einer Multiple-Choice-Aufgabe habe ich eine so genannte Führungsebene erstellt. Auf dieser Ebene wurden die dynamischen Texte erstellt. Diese dynamischen Texte werden die Werte „0“ oder „1“ bekommen. Am Anfang bekommen diese dynamischen Texte Wert „0“. Wenn eine vorgegebene Antwort ausgewählt wird, wird der entsprechende dynamische Text Wert „1“ bekommen. Im Gegenteil bleibt der entsprechende dynamische Text Wert „0“, wenn die vorgegebene Antwort nicht ausgewählt wird. Wenn die ausgewählte Antwort richtig wäre, wird wieder eine Richtig-Rückmeldung gegeben.

Der folgende Quellcode ist der ActionScript-Code der Aufgabe 1/4 beim Lernthema „Geometrie“, Kapitel „Spiegelachsen“.

```
d2a1_traurig_var = "0";
d2a2_traurig_var = "0";
d2a1_lach_var = "0";
d2a2_lach_var = "0";
//gesicht wird gezeigt
d2a1_btn.onRelease = function() {
    d2a1_lach_var = "1";
    d2a1_achsen2_mc._visible = true;
    d2a1_achsen_mc._visible = false;
    d2a1_achsen2_mc.gotoAndStop(2);
};
d2a2_btn.onRelease = function() {
    d2a1_traurig_var = "1";
    d2a2_achsen2_mc._visible = true;
    d2a2_achsen_mc._visible = false;
    d2a2_achsen2_mc.gotoAndStop(2);
};
d2a3_btn.onRelease = function() {
    d2a2_lach_var = "1";
```

```
        d2a3_achsen2_mc._visible = true;
        d2a3_achsen_mc._visible = false;
        d2a3_achsen2_mc.gotoAndStop(2);
};
d2a4_btn.onRelease = function() {
    d2a2_traurig_var = "1";
    d2a4_achsen2_mc._visible = true;
    d2a4_achsen_mc._visible = false;
    d2a4_achsen2_mc.gotoAndStop(2);
};
//
check_d2a_btn.onRelease = function() {
    if (d2a1_lach_var == "1") {
        d2a1_r_mc._visible = true;
    }
    if (d2a2_lach_var == "1") {
        d2a2_r_mc._visible = true;
    }
    if (d2a1_traurig_var == "1") {
        d2a1_f_mc._visible = true;
    }
    if (d2a2_traurig_var == "1") {
        d2a2_f_mc._visible = true;
    }
    if ((d2a1_lach_var == "1") && (d2a2_lach_var ==
"1") && (d2a1_traurig_var == "0") && (d2a2_traurig_var
== "0")) {
        _root.vogel_mc.gotoAndStop(2);
    }
    if ((d2a1_traurig_var == "1") || (d2a2_traurig_var
== "1")) {
        _root.vogel_mc.gotoAndStop(3);
    }
}
```

```
    }  
    if (((d2a1_lach_var == "1") && (d2a2_lach_var ==  
"0")) || ((d2a1_lach_var == "0") && (d2a2_lach_var ==  
"1"))) {  
        warnung1_mc._visible = true;  
    }  
    if ((d2a1_lach_var == "1") && (d2a2_lach_var ==  
"1")) {  
        warnung1_mc._visible = false;  
    }  
};
```

**Anmerkung:**

- d2a1\_traurig\_var, d2a2\_traurig\_var, d2a1\_lach\_var, d2a2\_lach\_var: die Variable-Namen den dynamischen Texten.
- d2a1\_btn, d2a2\_btn, d2a3\_btn, d2a4\_btn: die Instanznamen den Buttons den Achsen.
- d2a1\_achsen2\_mc, d2a1\_achsen\_mc, d2a2\_achsen2\_mc, d2a2\_achsen\_mc, d2a3\_achsen2\_mc, d2a3\_achsen\_mc, d2a4\_achsen2\_mc, d2a4\_achsen\_mc: die Instanznamen den MovieClips von Achsen.
- check\_d2a\_btn: der Instanzname des Überprüfens-Buttons.
- vogel\_mc: der Instanzname des Vogels.
- warnung1\_mc: der Instanzname des MovieClips vom Tipp „*Es gibt noch eine Spiegelachse. Versuch noch mal um sie zu finden!*“

## **6. Zusammenfassung und Ausblick**

Ziel dieser Masterarbeit war die mediendidaktische Konzeption und die prototypische Umsetzung einer Lernsoftware für Kinder. Das in der Arbeit beschriebene Programm basiert auf der Grundidee die Software nicht als reines Lernmedium, sondern als unterstützendes Hilfsmittel in den Mathematikunterricht Grundschule zu integrieren.

Die Arbeit wurde mit der Analyse den Vorteilen von Lernprogramm begonnen. In diesem Kapitel wurde ein Überblick über die Lernaktivitäten und die Nutzung von Lernprogramm durch Kinder gegeben. Das zweite Teil dieser Arbeit ist die Beschreibung des Programms „Stark in Mathe 3“, das ich für meine Masterarbeit entwickelt habe. Nach der Beschreibung des Programms wurde diese Arbeit mit dem Thema Didaktik weitergemacht. Das Thema Didaktik ist ein sehr weites Feld. Um dieses Thema genau zu analysieren, müssen verschiedene Aspekte berücksichtigt werden. Daher wurde mit der Definition der Zielgruppe begonnen, dann wurde das Thema mit den Analysen von Lehrziele und Lehrinhalte bzw. Lernmotivation vertieft. Auf diese Analysen wurden die Erkenntnisse der Mediendidaktik schrittweise aufgebaut. In diesem Teil der Arbeit wurde versucht, die lerntheoretischen Hintergründe und die darauf aufbauenden Konzepte der Mediendidaktik näher zu erörtern. Aufbauend auf diesen theoretischen Untersuchungen kam es dann zur Erstellung des Konzeptes für das Lernprogramm „Stark in Mathe 3“.

Nach dem Thema Didaktik hat das nächste Teil der Arbeit gezeigt, wie ich das Programm gestaltet habe. Das Design der Oberfläche und Buttons, die Navigationen und ihre Funktionen wurden in diesem Teil erklärt.

Bei der Bearbeitung des letzten Abschnittes wurde die technische Entwicklung beschrieben. Das Autorenwerkzeug des Programms Flash wurde in diesem Teil der Arbeit vorgestellt. Außerdem wurden die wichtigsten programmiertechnischen Umsetzungen auch in diesem Abschnitt erläutert.

Die Entwicklung des Programms „Stark in Mathe 3“ für Kinder hat mir viele neue Erkenntnisse und Ideen vermittelt, wie Spiel und Didaktik in einen multimedialen Rahmen integriert werden können. Die Erfahrungen mit *Macromedia Flash* als Autorenwerkzeug waren für mich außerordentlich positiv. *Flash* bietet einen sehr großen Spielraum bezüglich Gestaltung und Funktionalität von Lernobjekten.

## 7. Verzeichnis

### 7.1. Abbildungsverzeichnis

<i>Abb. 1: Struktur eines Inhaltsbereichs</i> .....	16
<i>Abb. 2: Struktur einer Aufgabe</i> .....	16
<i>Abb. 3: Eine Lückentext-Aufgabe</i> .....	18
<i>Abb. 4: Eine Multiple-Choice-Aufgabe</i> .....	18
<i>Abb. 5: Die Maus</i> .....	21
<i>Abb. 6: Startseite des Programms „Stark in Mathe 3“</i> .....	25
<i>Abb. 7: Bildschirmaufteilung eines Lernprogramms</i> .....	26
<i>Abb. 8: Bildschirmaufteilung des Programms „Stark in Mathe 3“</i> .....	27
<i>Abb. 9: Die Gefühlszustände des Vogels</i> .....	28
<i>Abb. 10: die gefühlsmäßigen Gesichter</i> .....	28
<i>Abb. 11: die Bleistifte aus der Software „Stark in Mathe 3“</i> .....	29
<i>Abb. 12: Screenshot der Flash-Entwicklungsumgebung</i> .....	33
<i>Abb. 13: Beispiel für ActionScript-Code</i> .....	33
<i>Abb. 14: Startbild der Bewegung</i> .....	35
<i>Abb. 15: Endbild der Bewegung</i> .....	35

### 7.2. Literaturverzeichnis

- **J. Peter Böhmer, Ekkehard Jander, Wilfried Schlake:** Mathematik begreifen 3. Schuljahr, Arbeitsheft – Teil 1: ISBN 3-12-166771-8 und Teil 2: ISBN 3-12-166772-6. Ernst Klett Grundschulverlag.
- **Jakob Nielsen:** Designing Web Usability. Erfolg des Einfachen. ISBN 3-8272-6846-X. Markt + Technik.
- **Caroline & Matthias Kannengiesser:** Flash 8 Espresso! ISBN 3-7723-7106-X, Frazis Verlag GmbH, 2006.
- **Friedrich Hagedorn:** Beitrag „Lernen im Netz: Überschätzt oder Aufbruch in neue Welten?“, S.1.

- 
- **BITCOM** - <http://www.checkpoint-elearning.de/article/4625.html>. Stand 15.04.2008.
  - **Medienpädagogischer Forschungsverbund Südwest: KIM – Studie 2006** – download unter: <http://www.mpfs.de>.
  - **Werner Renz:** Beispielaufgaben für Vergleichsarbeiten im Fach Mathematik in der Klasse 3 – Freie und Hansestadt Hamburg, Behörde für Bildung und Sport Amt für Bildung.
  - **Torsten Rülke:** Anwendung objektorientierter Konzepte mit ActionScript 2 zur Realisierung wieder verwendbarer Lerninhalte – Design und Entwicklung des Kurses „Die interaktive Arbeit mit Flash MX 2004“ für das Bildungsportal Sachsen. Diplomarbeit an der Hochschule für Technik und Wirtschaft Dresden, 2006.
  - **Swen Jürgens:** Evaluation von world-wide-web basierten Benutzungsschnittstellen für Kinder. Diplomarbeit im Fach Informatik an der Universität Hamburg, 2003.
  - <http://www.tu-chemnitz.de/phil/elearning/studentenprojekte/motivation/arcs.htm>.
  - <http://de.wikipedia.org/>

## **Erklärung**

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Braunschweig, den 15.06.2008

Nguyen Nam Khanh