



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

## ***Zwischenbericht***

Robot - Building - Lab

vorgelegt von

***Inga Mähl, Werner Baum, Gunter Klemke, Christine Klemm, Kai von Luck, Kai Rosseburg, Sven Tennstedt***

am 16.06.2007

finanziert durch die Müller-Reitz-Stiftung

durchgeführt von

**Hochschule für Angewandte Wissenschaften Hamburg Department für Informatik; Luisengymnasium Hamburg; Kurt - Körber - Gymnasium Hamburg**

## **Ein Robot - Building - Lab**

Die Idee ein Robot - Building - Lab an Schulen einzurichten entstand aus unterschiedlichen Gründen. Zum einen ist es eine gute Möglichkeit spielerisch fachliches Grundwissen zu schaffen, zum anderen trägt es die Hoffnung weitreichende Auswirkungen auf spätere Berufsfindungen der Schüler/-Innen zu haben.

Das Robot - Building - Lab verbindet vielseitige Arbeitsbereiche und gibt einen Überblick wie sich projektorientierter Unterricht, bzw. ein begleitetes Projekt, positiv auf den Kompetenz- und den Wissenserwerb von SchülerInnen auswirkt. Der handlungsorientierte Ansatz dieses Vorhabens bietet den Vorteil der kreativen Gestaltung des Informatikunterrichts. Selbstständige Analyse, Modellierung, die Entwicklung und Beschreibung eigener Lösungsansätze, sowie deren Reflexion und Verbesserungen fördern kreatives und schöpferisches Denken. Die Schüler/-Innen arbeiten ein Schuljahr lang in Kleingruppen mit einem Roboter, wobei sie verschiedene vorgegebene und eigenständig erarbeitete Aufgaben lösen. Dabei wird eine neue Form des Unterrichtens getestet, deren Vorteil in der Kombination von traditionellen, wie z.B. Frontalunterricht, und neueren Unterrichtsformen, z.B. selbstgesteuertes Lernen, liegt. Der Unterricht erhält einen explorativen Charakter, da er sich weitgehendst auf den Lernenden stützt. In lernerzentrierten Umgebungen kann mehr Engagement festgestellt werden und die Teilnehmer lernen intrinsisch motiviert. Zudem reflektieren die Schüler/-Innen das zu Erlernende und ihre eigenen Fähigkeiten. Kann Schüler/-Innen ein Basiswissen über ihre Fähigkeiten und bestimmte Grundmuster von Methoden und Strategien vermittelt werden, so haben sie einen weitreichenden Vorteil gegenüber anderen.

Informatik ist eine Wissenschaft, die sich stetig und schnell entwickelt. Demzufolge treffen die Schülerinnen und Schüler in ihrem Unterricht auf sich wiederholende Konzeptionen und Wirkungsprinzipien, die ihnen ermöglichen sich rasch in die Nutzung neuer Systeme einzuarbeiten und erlangen damit eine Flexibilität, die weitere berufsbezogene Vorteile schafft.

Neben diesen genannten Erfahrungen und Vorteilen, haben die Schüler/-Innen sehr viel Spass im Unterricht, weil sie spielerisch ihre Aufgaben lösen und dabei ständig neue Ideen entwickeln.

Ein besonderer Dank gilt der Müller - Reitz Stiftung, die dieses Projekt finanziert hat.

# Inhaltsverzeichnis

<b>1 Projekt</b>	<b>5</b>
<b>2 Informatik</b>	<b>7</b>
Didaktische Ziele . . . . .	11
Umsetzung und Probleme . . . . .	12
Weiterführende Ideen . . . . .	12
<b>3 Institutionen</b>	<b>14</b>
<b>4 Vorbereitung</b>	<b>19</b>
Unterricht: . . . . .	21
<b>5 Technik</b>	<b>23</b>
<b>6 Der Unterricht am Luisengymnasium</b>	<b>27</b>
<b>7 Der Unterricht am Kurt-Körper-Gymnasium</b>	<b>36</b>
<b>8 Projektablauf</b>	<b>39</b>
<b>9 Bewertung des Ansatzes</b>	<b>45</b>
<b>10 Ausblick</b>	<b>48</b>
<b>A Material</b>	<b>50</b>
<b>Literaturverzeichnis</b>	<b>61</b>

# Abbildungsverzeichnis

4.1	Rescue League . . . . .	20
5.1	Legoumgebung . . . . .	23
5.2	Entwicklungsumgebung C# . . . . .	25
6.1	Lego NXT - konstruiert von Schülern . . . . .	27
6.2	Robotersimulation . . . . .	28
6.3	weiterer NXT, konstruiert von SchülerInnen . . . . .	31
6.4	Schüler mit NXT . . . . .	33
6.5	NXT konstruiert von SchulerInnen . . . . .	33
7.1	SchülerInnen des Kurt-Körper-Gymnasiums . . . . .	37
9.1	Präsentation des Kurt-Körper-Gymnasiums . . . . .	46
9.2	SchülerInnen präsentieren auf der Nacht des Wissens . . . . .	47
10.1	Gäste der Nacht des Wissens . . . . .	49



# 1 Projekt



Der vorliegende Bericht stellt eine erste Zwischenbilanz des Projektes „Robot Building Lab“ dar, das mit Mitteln der Müller-Reitz-Stiftung seit Sommer 2006 finanziert wird. Ziel dieses Projektes ist es, dass Schülerinnen und Schüler der Mittelstufe spannendes naturwissenschaftliches und technisches Wissen anwenden und Begeisterung zeigen für Fachfragen: Sie entwickeln und bauen autonome Roboter, die sie in selbst definierten Umgebungen interagieren lassen. Seit 1999 gab es von engagierten Professoren an der HAW Hamburg einzelne Kooperationen mit bestimmten Klassen im Informatik- und Physikunterricht, abhängig von der jeweiligen Lehrperson. Das Neue an dieser Projektidee ist die Verstärkung der Zusammenarbeit von Schülerinnen und Schülern, Studierenden und den Lehrenden an Schule und Hochschule mit dem Ziel, dass ein „Robot-Building-Lab“ selbständig und nachhaltig von den Schulen eingesetzt werden kann. Im Gegensatz zu den bisherigen Ansätzen, ein solches Programm als gesondertes Projekt durchzuführen, ist es Ziel dieses Ansatzes, das Robot-Building-Lab in den regulären Unterricht der Sekundarstufe I zu integrieren. Grundlage dieser Projektidee ist die Einrichtung eines mobilen Robot-Building- Labors. Es soll im Einsatz in einem Gymnasium erprobt werden. Die Übertragbarkeit der Ergebnisse wird dann in einer zweiten Phase in einem zweiten Gymnasium überprüft und könnte dann als Modell auch auf weitere Schulen übertragen werden. Diese Roboterlabore haben weltweit eine gewisse Tradition, Themen des Maschinenbaus, der Elektrotechnik und der Informatik zu einem ganzen zu integrieren (als interdisziplinäre Disziplin auch als Mechatronik bekannt). Als Anwendungen werden neben klassischen Anwendungen inzwischen auch Themen der Robot-Kunst ,z.B. [http://users.informatik.haw-hamburg.de/~lego/Projekte/f18/f18\\_deutsch.html](http://users.informatik.haw-hamburg.de/~lego/Projekte/f18/f18_deutsch.html), an-

geboten, die eine eher nicht-technische Komponente beinhalten. Die Arbeit mit dem Roboterlabor erfordert eine kontext- und handlungsorientierte Didaktik, wie projektorientiertes Lernen, vernetztes (ganzheitliches) Denken, exploratives Lernen, Anleitung zur Modellierung komplexer dynamischer Systeme und ziel- und terminorientiertes Lernen.

Nicht nur technikbegeisterte Jungen bauen Roboter, sondern Mädchen sind über Roboter-Projekte für Mathematik, Informatik, Naturwissenschaft und Technik (MINT) zu interessieren, wie neben eigenen vielfältigen Erfahrungen es z.B. auch das Roberta-Projekt (Mädchen erobern Roboter) des Fraunhofer Instituts für Autonome Intelligente Systeme gezeigt hat, [ais.fraunhofer.de](http://ais.fraunhofer.de)<sup>1</sup>. Mädchen und Jungen können ihre Begeisterung für die Roboter in nationale und internationale Wettbewerbe einbringen.

---

<sup>1</sup> (fraunhofer:2005, 2005)

## 2 Informatik

### Rahmenplan Informatik

Im Rahmenplan für das Wahlpflichtfach Informatik der Behörde für Bildung und Sport in Hamburg sieht das Hauptziel des Unterrichtsfaches folgendermaßen:

„Der Informatikunterricht hat das Ziel, Schülerinnen und Schülern einen Zugang zur Nutzung, Analyse, Beschreibung und Modellierung komplexer Informationsstrukturen aus Wissenschaft, Wirtschaft und Gesellschaft zu ermöglichen.“<sup>1</sup>

Der Nutzen des Informatikunterrichtes setzt sich aus verschiedenen Bereichen zusammen. So soll ein Verständnis der Wirkungsweisen von modernen Informations- und Kommunikationstechniken durch das Praktizieren von informatischer Denkweise geschult werden. Innerhalb der einzelnen Aufgabenstellungen, wie zum Beispiel der Konstruktion von Informatiksystemen oder Teilen davon, werden Problemlösungsmethoden erarbeitet, die im späteren Alltag, Wirtschaft oder Wissenschaft Vorteile schaffen können. Damit soll eine gleichberechtigte Teilhabe am gesellschaftlichen Leben und einem lebenslangem Lernen begründet werden.

### Erwerb von Grundlagenkompetenzen

Der Informatikunterricht vermittelt den Schülerinnen und Schülern ein Orientierungswissen anhand dessen sie elektronisch verfügbare Informationen leichter erschliessen, strukturieren und erarbeiten können. Dieses Wissen stärkt die Handlungsfähigkeit in diversen Problemsituationen. Die erworbene Methodenkompetenz der Schülerinnen und Schüler hilft ihnen sich in den komplexen und vernetzten Systemen zurecht zu finden.

Informatikunterricht stellt einen Grundbaustein in der Medienkompetenz, da in diesem Fachbereich die Wechselwirkung und die Struktur von Informatiksystemen analysiert wird und die Teilnehmer eigene mediale Produkte und Systeme gestalten sollen. In all den genannten Punkten ist ein fächerübergreifender Unterricht immanent vorhanden, da hier die Werkzeuge zur Strukturierung und Verknüpfung von verschiedenen Informationen bereitgestellt

---

<sup>1</sup> (Freie und Hansestadt Hamburg, 2003), S.5

werden. Die anwendungsorientierte Nutzung vom erworbenen Wissen fördert die Wahrnehmung, Durchdringung und Bearbeitung von komplexen Problemstellungen. Informatik ist eine Wissenschaft, die sich ständig und schnell entwickelt, demzufolge treffen die Schülerinnen und Schüler in ihrem Unterricht auf sich wiederholende Konzeptionen und Wirkungsprinzipien, die ihnen ermöglichen sich rasch in die Nutzung neuer Systeme einzuarbeiten. Neben den handlungsorientierten Vorteilen des Faches bietet die Informatik kreative Gestaltungsmöglichkeiten. Selbstständige Analyse, Modellierung, die Entwicklung und Beschreibung eigener Lösungsansätze, deren Reflexion und Verbesserungen fördern kreatives und schöpferisches Denken.

## **Gruppen-und Projektarbeit**

Durch die komplexen Problemstellungen der Aufgaben und deren Lösung innerhalb eines bestimmten Zeitraumes ist ein arbeitsteiliges Vorgehen nötig. Sie lernen so, dass Kommunikation und Kooperationsfähigkeit wesentlich sind, um ein Projekt oder eine Aufgabe zu gestalten. Die Durchführung und Lösung einer Aufgabe durch Gruppenarbeit führt sie an Projektarbeit und die damit verbundenen Denkansätze hinsichtlich Planung, Entscheidungen treffen, Präzision, Aufteilung von Aufgaben und Einhaltung von Absprachen heran.

## **Berufsorientierung**

Der Unterricht hilft den Teilnehmern sich in der Berufswelt zu orientieren und vermittelt ihnen einen Vorsprung. Sie erhalten wesentliche Grundlagen in Kommunikations- und Informationstechniken und Erfahrungen in typischen Problemlösungen, die ihnen die notwendige Basis für diverse Berufsfelder geben. Desweiteren sollen die Schülerinnen und Schüler befähigt werden ihre erlernten Denk- und Arbeitsweisen verantwortungsbewusst einzusetzen. Der Einblick in die unterschiedlichen IT Berufe ermöglicht den Teilnehmern eine erweiterte Berufswahl und neue Möglichkeiten auf dem Arbeitsmarkt zu erkennen. So wird es im Rahmenplan für das Wahlpflichtfach Informatik in der Sekundarstufe 1 beschrieben. Was genau steckt dahinter? Schülerinnen und Schüler sollen befähigt werden verantwortungsbewusst mit dem Medium Computer umzugehen. Diese Kompetenz soll durch unterschiedliche Inhalte und Themen erreicht werden. Der Rahmenplan Informatik orientiert sich an vier Leitlinien.

## **Leitlinien**

Die Schülerinnen und Schüler sollen in *Interaktion mit Informatiksystemen* treten, das bedeutet sie arbeiten sich zunehmend selbstständig in die Nutzung von Informatiksystemen

ein, indem sie in globalen Informationsräumen recherchieren und navigieren. Dabei werden Bewertungskriterien für die verwendete Software, Methoden und Strategien erarbeitet, die ihnen bei späteren Lösungsansätzen hilfreich sein können. Die *Wirkprinzipien von Informatiksystemen* stellen eine weitere Leitlinie für den Informatikunterricht dar. Hier lernen die Teilnehmer die Bestandteile von Informatiksystemen kennen, die Zusammenwirkung von Systemkomponenten und die Einordnung der Teilsysteme in Systemzusammenhänge. Zudem wird der prinzipielle Aufbau von komplexen Informatiksystemen, Prinzipien, Verfahren und Algorithmen erlernt, die die Basis der erworbenen Kenntnisse von Ideen und Konzepten der Informatik ergänzen. Dies soll den Schülerinnen und Schülern zeigen, dass die Komplexität von Information reduziert werden kann, sowie ihre geistigen Fähigkeiten gestärkt werden.

Schülerinnen und Schüler erkennen im Informatikunterricht Problemsituationen, lernen diese zu analysieren, zu *modellieren* und ein dementsprechendes Modell nach den schulischen Gegebenheiten zu implementieren. Hierfür lernen sie Modellierungstechniken kennen, die ihnen innerhalb sowie ausserhalb der Schule nützlich sein werden. Die Teilnehmer erkennen, dass jedes Informatiksystem auf der Grundlage eines vereinfachten Abbildes der Realität basiert. Als letzte Leitlinie für den Informatikunterricht wird die *Wechselwirkung zwischen Informatiksystem, Individuum und Gesellschaft* gesetzt.

In diesem Bereich wird Verantwortungsbewusstsein für Technik geschult. Das Verhältnis Mensch - Technik wird geschichtlich und gesellschaftlich untersucht, damit die Schülerinnen und Schüler die Möglichkeiten, Grenzen und Gefahren von Technik beurteilen können und die Auswirkungen auf die Gesellschaft verstehen. Sie erhalten Kenntnisse über die (Aus-)Wirkungen von dem Einsatz von Informatiksystemen auf die Gesellschaft und Individuen, damit können sie Kriterien entwickeln, die eine menschengerechte und sozialverträgliche Technik ermöglicht.

Der Rahmenplan sieht den Schwerpunkt im Unterricht darin möglichst weitreichende Kompetenzen zu schaffen, um den Schülerinnen und Schülern vielseitige Einblicke in Berufsfelder zu geben und Qualifikationen für den Arbeitsmarkt zu schaffen, damit sie einen Vorteil bei der Arbeitssuche bekommen. Es soll ein gutes Allgemeinwissen vermittelt werden, weiterführende Kenntnisse können im späteren Ausbildungen angeeignet werden. Die Schülerinnen und Schüler sollen eine Vorstellung von bestimmten „Mustern“ in der Informatik erhalten, damit sie später durch Wiedererkennung von bestimmten Strukturen einen Vorteil bei Problemlösungen erhalten und so der Umgang mit den neuen Medien erleichtert wird.

## Lehrzentrierte Einflüsse

Soweit zum Rahmenplan und seinen Anforderungen an den Informatikunterricht in der Sekundarstufe 1. Wie sehen Lehrerinnen und Lehrer ihren Unterricht? Welche Anforderungen

stellen sie an den Unterricht und die zu vermittelnden Informationen?

Viele Lehrer/-Innen möchten ihren Klassen ein gutes Allgemeinwissen vermitteln anhand dessen sie auf spätere Probleme und Aufgabenstellungen vorbereitet werden, ähnlich wie es im Rahmenplan vorgesehen ist. Der Unterricht wird auf das spätere Leben und dessen Anforderungen ausgerichtet. Welche Vorteile bietet welches Wissen den Schülerinnen und Schülern im späteren Leben? Welche Fakten können bei Bewerbungen den Lebenslauf aufwerten, um eine bessere Chance auf dem Arbeitsmarkt zu haben? So wird oftmals mit bestimmten Produkten gearbeitet, die möglichst weitverbreitet sind, damit die Schülerinnen und Schüler grössere Möglichkeiten bekommen.

Im Unterricht wird mit bestimmten Produkten/Anbietern gearbeitet, die einen besonderen Reiz oder Zusammenhang zu den Schülerinnen und Schülern bilden, um sie für ein Thema zu interessieren. Den Klassen wird deshalb eher ein Produktwissen vermittelt, da so schnell ein Erfolgserlebnis für die Schülerinnen und Schüler erreicht wird, ein „da kann ich was“ Ausdruck, da sie sich schnell zurechtfinden. Die Lehrkräfte möchten sich durch das Einsetzen von Produktwissen an die Konzepte der Informatik herantasten.

Würde der Unterricht rein auf Konzepten ohne jeglichen direkten Bezug zu den Teilnehmern basieren, würde die Klasse sich nicht engagieren und keine neuen Erkenntnisse aus dem Unterricht für sich gewinnen können. Bei vielen Lehrkräften kann eine Verunsicherung in der Gestaltung des Unterrichtes festgestellt werden. Zum einen sind Themen, die Schülerinnen und Schüler interessieren könnten in der Schule verboten, wie zum Beispiel die Benutzung von Handys. Doch könnten sich in diesem Themenbereich vielzählige Unterrichtseinheiten ansiedeln, die jedoch nicht durchgeführt werden, da an der Schule ein Handyverbot gilt. Hinter vielen dieser ablehnenden Ausführungen steckt die Befürchtung, weniger wissen zu können als die Schulklasse. Lehrerinnen und Lehrer müssten ihr sicheres Terrain verlassen und könnten eventuell nicht auf alle entstehenden Fragen eine Antwort wissen. Normalerweise nimmt der Lehrer/-Innen eine führende Rolle im Unterricht ein, er lenkt den Unterricht, gibt Lernziele vor und verfolgt seine Zielsetzungen. Es ist schwer diese Rolle abzulegen und zusammen mit der Klasse zu lernen. In diesem Zusammenhang spannt sich die Diskussion um selbstgesteuertes Lernen an Schulen, darauf werde ich später weiter eingehen.

Einige Lehrer/Innen argumentieren mit Aussagen wie „Unsere Schüler schaffen eben nicht mehr“, wenn sie die Arbeit mit Produktwissen begründen möchten, andere damit, dass Schülerinnen und Schüler übertragbares, allgemeines Wissen benötigen, die Arbeit mit weitverbreiteten Produkten hilft ihnen sich zurecht zu finden.

Es gibt Kritik an dieser Art der Wissensvermittlung. Kritik an dieser Arbeit mit reinem Produktwissen speist sich aus der Kurzlebigkeit von dieser Art des Wissens, das es oft mal rein auswenig gelernt wird, isolierte Fakten gelernt werden und es an bestimmte Produkte gebunden ist. Konzeptwissen stellt das Gegenteil dar, es ist dementsprechend langlebig, basiert auf Verständnis und der Darstellung von Zusammenhängen, seine Produktunabhängigkeit bietet den Vorteil, dass der gewünschte Wiedererkennungswert von

Lösungsmustern leichter übertragbar ist.

Im Unterricht werden meistens Anwendungstechniken gelehrt, die Schülerinnen und Schüler sollen „fit gemacht werden“ im Umgang mit den neuen Medien und dem Internet. Hintergrundwissen wird selten ausgebildet, da an Schulen „keine Programmierer ausgebildet werden sollen“. Mehr Informationen und Fähigkeiten können im späteren Berufsleben erworben werden. Inwieweit ein besseres Hintergrundwissen an Schulen zu einem weit-gefasstem Verständnis der Informatik beitragen kann, müsste in diesem Zusammenhang weiter durchdacht werden.

Im Rahmenplan für Informatik stellt der Bereich Reflexion einen wichtigen Baustein dar. Im Unterricht wird dementsprechend über die Gefahren des Internets diskutiert, diese Gespräche werden nicht auf allgemeine oder detaillierte Technik ausgebreitet, sondern verlaufen meistens in dem genannten Gebiet.

Begriffe der Informatik werden im Unterricht nur eingesetzt, wenn es unbedingt nötig ist, dann überwiegend im Zusammenhang mit Grafikeinheiten, weniger mit Textarbeit, die beide im Unterricht eine grosse Rolle einnehmen. Dabei ist Textverarbeitung der Bereich mit dem die meisten Klassen an den Informatikunterricht herangeführt werden.

Der Schwerpunkt an Schulen liegt darin ein Grundwissen zu schaffen, Fertigkeiten und Fähigkeiten in geläufigen Systemen und Programmen zu vermitteln, die dem Lebenslauf zweckmässig angepasst werden. Ein mehr oder weniger ausgeprägtes Hintergrundwissen ist für diesen Zweck nicht ausschlaggebend. Es wird ein rein anwendungsorientiertes Wissen vermittelt. Weiterführende Kenntnisse können im späteren Lebensverlauf der Schülerinnen und Schüler erworben werden, entweder in beruflicher Hinsicht oder Privater.

## **Didaktische Ziele**

Der Informatikunterricht stellt Lernsituationen in denen anwendungsbezogene Problemstellungen im Mittelpunkt stehen, die sich aus dem Bereichen Berufswelt, Wissenschaft, Wirtschaft, Verwaltung oder dem Erfahrungsbereich der SchülerInnen zusammensetzen. Demnach ein sehr weitreichendes Gebiet der Möglichkeiten. Die Problemstellung soll exemplarisch sein und sich auf typische Anwendungsgebiete, zum Beispiel Text-Dokumente, Grafik, Kommunikation, Präsentation oder Simulation, beziehen. Der fächerübergreifende Charakter der Aufgaben ist in jedem Bereich wünschenswert.

Der Unterricht orientiert sich an den jeweiligen Handlungsmöglichkeiten der Lernenden, die eigenständig Untersuchungen und Analysen des geforderten Sachverhaltes durchführen. Dabei können SchülerInnen die benötigten Methoden der systematischen Problemstrukturierung nur dann erfassen und lernen, wenn sie diese Strukturierung selbst vollziehen. Bei der Lösung von Aufgaben werden u.a. Modelldiagramme entworfen und selbsttätig der Umgang mit Informatiksystemen geübt. Der Lernende soll eigenständig in den Bereichen Informati-

ongewinnung, -strukturierung und -präsentation Fähigkeiten erwerben, die dafür benötigte Motivation ist in der Aufgabenstellung und durch den Lehrer/-Innen zu finden.

Teamarbeit, Selbstständigkeit und Kommunikation sind drei wesentliche Teilbereiche, die durch typische, projektbezogene Aufgabenstellungen erlangt werden sollen. Jede Aufgabenstellung soll offen gehalten werden. Dies erfordert zunächst eine Anforderungsanalyse und die Entscheidung über mögliche Beschränkungen und die Zerlegung in Teilprobleme.

Die in diesem Projektabschnitt angewandte Aufgabenstellung verbindet diese Anforderungen des Lehrplans an den Informatikunterricht, da die Simulationsgestaltung zu einen im Alltag der SchülerInnen zu finden ist, weil sich der Grossteil des Klassenverbandes mit Computerspielen auskennt, und zum anderen werden reale Bezüge in der Aufgabenstellung verwendet.

## Umsetzung und Probleme

Die Unterrichtsgestaltung ist zum Grossteil an die Erfahrungen des Unterrichtenden gekoppelt. Hier liegt eine weitere Besonderheit des Informatikunterrichtes. Viele der Lehrer und Lehrerinnen beziehen ihre Kenntnisse in diesem Fach aus Fortbildungen oder Schulen sich selbst. Zur Zeit gibt es sehr wenige LehrerInnen, die dieses Fach erlernt oder studiert haben. Dementsprechend gibt es Hemmschwellen in der Gestaltung des Unterrichtes. So ist nicht nur die Angst davor weniger Kenntnisse zu besitzen als die SchülerInnen, sondern auch die Befürchtung in eine fachliche Sackgasse zu geraten ein grosses Hindernis, um neue Elemente in den Unterricht einzufügen. Vereinfacht werden könnte diese Problematik, wenn der fächerübergreifende Unterricht vertieft wird. Da es für den Lehrplan der Informatik keine festen Vorgaben gibt, ist die Lehrkraft an keine Richtlinien gebunden. Eine einzige Hürde stellt die zentrale Abschlussprüfung dar, die SchülerInnen, die sich in den Oberstufen befinden, müssen über bestimmte Grundlagen verfügen, um einen guten Abschluss zu erhalten. Diese Freiheiten, die die Gestaltung des Lehrplans mit sich bringt, können weitreichend genutzt werden, um die Interessenlagen der Schulklassen miteinzubeziehen und andere Unterrichtsformen auszuprobieren, um fundierte Kenntnisse zu vermitteln.

Die Begriffe des Lehrplans sind zum Teil nicht deutlich dargestellt werden. Probleme über den Inhalt des Unterrichts ergeben sich hier, wenn die/die unterrichtende/n LehrerIn nicht als Informatiklehrer ausgebildet ist. Es fehlen Vorstellungen über Möglichkeiten und Hintergründe, was diese Begriffe alles beinhalten können.

## Weiterführende Ideen

*Projektorientierter Unterricht:* Durch projektartigen Unterricht kann nicht nur den SchülerInnen das Thema näher gebracht werden, sondern die Lehrkraft kann sich freier bewegen,



da sie mit den SchülerInnen zusammen lernen kann. Im Frontalunterricht müsste er das gesamte Aufgabengebiet direkt vermitteln, so kann er sich konkret um Besonderheiten kümmern und die Klasse beim Lernen betreuen. Anhand der Grenzen, die sich in dieser Art von Unterricht auf tun könnten, können die Schwierigkeiten in der Informatik verdeutlicht werden und so zum Bestandteil des Wissenserwerbs werden. Praxisbezug im Unterricht verdeutlicht nicht nur Probleme des zu behandelnden Unterrichtsstoffes, sondern gibt den SchülerInnen Selbstsicherheit im Umgang mit Schwierigkeiten und neue Lösungsansätze bei möglichen späteren Problemen. Dies würde mit dem Rahmenplan für Informatik als Wahlpflichtfach konform sein.

Die Schülergruppen bilden sich eigenständig, wobei wichtig ist, dass maximal 3 SchülerInnen in einer Gruppe sind, da sonst ein intensives Arbeiten nicht möglich ist. Idealerweise arbeiten 2 SchülerInnen zusammen, da sie sich so in der Programmierung und der Beobachtung abwechseln können. Diese Art der Zusammenarbeit lässt sich in der Informatik unter dem Begriff „pair programming“ (Beck, 2000) wiederfinden. Hier arbeiten 2 Personen zusammen, wobei sich die Teilnehmer ständig im Programmieren und Beobachten abwechseln.

# 3 Institutionen

## Hochschule für Angewandte Wissenschaften

Das Roboter-Labor an der Informatik der HAW wurde von Prof. Dr. G. Klemke und Prof. Dr. K. v. Luck 1997 gegründet und führt u.a. Projekte in Kooperation mit dem Department Maschinenbau der HAW Hamburg, der Informatik und der Mechatronik der FH Brandenburg sowie der Informatik der FH Dortmund durch. Es entstanden seit Gründung des Labors über 50 Studien- Diplom- und Bachelorarbeiten. (s.a. <http://users.informatik.haw-hamburg.de/kvl/papers.html>)

Seit 1999 führt das Robotik-Labor projektartige Schulkurse mit Lego-basierten Robotern sowohl für Schüler und Schülerinnen als auch für Lehrer und Lehrerinnen (Multiplikatorschulungen) durch. Eine erste Auswertung der gewonnenen Erfahrungen erfolgte durch Matthias Stolt (Roboter im Informatikunterricht, Studienarbeit HAW Hamburg 2001)

Das Roboter-Labor der Informatik der HAW Hamburg war in den Jahren 2002, 2003 und 2004 Gastgeber des Regionalwettbewerb FirstLegoLeague (ein internationaler Robotik-Wettbewerb), in Deutschland teilweise finanziert durch hands-on-technology e.V. (s.a. [www.firstlegoleague.org/de](http://www.firstlegoleague.org/de))

Seit Herbst 2005 findet eine Kooperation mit der Firma Still, des Maschinenbaus der Helmut Schmidt Universität und dem Gymnasium Billstedt im Projekt e-truck statt. Hierbei werden Roboter durch gemischte Teams von Schülern und Lehrlingen, begleitet durch die beteiligten Hochschulen, konstruiert. Dieses Projekt wird finanziert durch die Körberstiftung.

Das Roboter-Labor der Informatik der HAW Hamburg ist eines von ca. hundert Laboren weltweit, die eine Vorversion des neuen Lego-Mindstorm-Konzepts von Lego zur Evaluation zur Verfügung bekommen hat, um auf dieser neuen Plattform neue Konzepte zu entwickeln. Die endgültige Version ist inzwischen erhältlich und stellt auch die Grundlage für dieses Projekt darstellen.

Die Informatik der HAW Hamburg hat mit Prof. Dr. K. v. Luck eine Kooperation mit der Informatik der Uni Hamburg im Rahmen des Schul-Support-Systems. In diesem Projekt erfolgt eine Rechnerwartung an über 120 Hamburger Schulen durch Informatik-Studierende. Dieses Projekt stellt somit einen engen Kontakt zwischen den beteiligten Hochschulen und der Informatik-Lehrern und -Lehrerinnen her und wird finanziert durch die Behörde für Bildung und Sport in Hamburg.

Die Hochschule für Angewandte Wissenschaften Hamburg sieht in ihrem anwendungsbezogenen Studienangebot eine Chance, einerseits ihre Zusammenarbeit mit Schulen nachhaltig zu gestalten und andererseits ihren Beitrag dazu zu leisten, mehr Schüler und insbesondere Schülerinnen für ein technisch-naturwissenschaftliches Studium zu gewinnen. Für das Department Informatik gilt dieses insbesondere, da schon aus den bestehenden Aktivitäten ein stetiger Bewerberstrom auch aus ehemaligen TeilnehmerInnen solcher Kurse zu verzeichnen ist. Somit ist die kontinuierliche Zusammenarbeit der Informatik der HAW mit Gymnasien (Organisation von Informatik-Fachlehrertreffen, Beteiligung am Schulsupport der Schulbehörde, Ausrichtung der FirstLegoLeague, Informationsveranstaltungen in Schulen und vieles andere) eine gute Tradition, in die sich die Robot-Building-Labs einfügen als Teil eines Gesamtkonzepts von Wissenschaftsmarketing.

### **Das Luisengymnasium**

Das Luisengymnasium befindet sich in einem ruhige Wohngebiet in unmittelbarer Nähe zum Bergedorfer Gehölz. In dem von dem Hamburger Architekten Fritz Schumacher entworfenem Gebäude lernen zur Zeit ca. 900 Schülerinnen und Schüler. Die Schule zeichnet sich durch besondere Bildungsbereiche aus, so gibt es einen Musikzweig ab der 5. Klassenstufe, mit dem besonderen Schwerpunkt Orchester, einen naturwissenschaftlichen - technischen Wahlbereich und es ist ein Aufbaugymnasium. Schülerinnen und Schüler mit einem Real-schulabschluss können nach Absolvierung des Eingangsjahres in die Oberstufe übergehen und ihre schulische Ausbildung mit einem Abitur abschließen. Die Naturwissenschaften sind ein wichtiges Element am Luisengymnasium. Jedes Fach nähert sich dabei auf unterschiedliche Weisen spezifischen Themen an, wobei Bewußtsein, Respekt und Verständnis für Natur und Umwelt geschaffen wird. Eigenes Experimentieren, Reflexion der Experimente oder Diskussionen von Demonstrativexperimenten sind wesentliche Elemente des Unterrichts. Das Schülerexperiment stellt ein wichtiges Element des Unterrichtes in der Physik, der Chemie und der Biologie dar. Im Fach „Natur und Technik“ im Wahlbereich der Klassenstufen 5 bis 8 und das „naturwissenschaftliche Praktikum“ als Wahlpflichtfach in den Klassen 9 und 10 bietet die einzigartige Möglichkeit, sich mit naturwissenschaftlichen und technischen Fragestellungen durch praktisches Experimentieren eingehender auseinanderzusetzen.

Zudem wird mit dem Cassy-System, ein computergestütztes Messwerterfassungssystem, gearbeitet. In den folgenden Jahrgangsstufen wird als Ergänzungskurs ein zusätzlicher naturwissenschaftlicher Experimentierkurs angeboten und die Zusammenarbeit im sogenannten „Bergedorfer Modell“ ermöglicht. In jedem Jahr ist die Wahl von Grund- und Leistungskursen in allen naturwissenschaftlichen Fächern bis zum Abitur möglich. Nach der Einführung der Profiloberstufe zum Schuljahr 2009/2010 wird weiterhin ein naturwissenschaftlich-informationstechnisches Profil angeboten.

Das Luisen-Gymnasium ist, wie bereits erwähnt, dabei ein naturwissenschaftlich-technisches Profil zu entwickeln und zu vertiefen. Aus diesem Grund werden zu den bisher angebotenen Themen aus den Bereichen Biologie, Physik und Technik nun auch Inhalte der Chemie und der Informatik treten. Um diesen letzten Bereich für Schüler reizvoll zu gestalten, soll das Praktikum in Form eines Roboterlabors durchgeführt werden.

Zudem wird zu diesem Projekt mit der Behörde eine Ziel- und Leistungsvereinbarung abgeschlossen, um diesem Projekt in der Schulentwicklung des Luisen-Gymnasiums einen außerordentlichen Stellenwert einzuräumen.

## **Das Kurt -Körper-Gymnasium**

Das Kurt-Körper-Gymnasium wurde 1968 als Gymnasium Billstedt gegründet und im Januar 2007 zum Kurt - Körper - Gymnasium umbenannt. Das Gymnasium gehört zu den Pioniersschulen, die als erste Informatik in ihr Unterrichtsangebot aufgenommen und 1982 mit Apple-Rechnern begonnen haben. Schon 1993 wurde ein Netzwerk für den Unterricht in Informatik und ITG eingerichtet. Die Schülerschaft besteht zu etwa zwei Dritteln aus Schülerinnen und Schülern mit Migrationshintergrund. Zum Einzugsgebiet gehören z.B. die 1994 errichteten Dörfer zur Aufnahme von Aussiedlern aus ehemaligen Sowjetrepubliken wie Kasachstan und aus Polen.

Als zentrale Aufgabe unserer gemeinsamen Arbeit sehen wir, die Mitglieder der Schulgemeinschaft des Kurt-Körper-Gymnasiums, die Entwicklung und Förderung der Gestaltungskompetenz unserer Schülerinnen und Schüler im Hinblick auf zukünftige Aufgaben und Anforderungen. Die von uns angestrebte Gestaltungskompetenz orientiert sich dabei an drei übergreifenden Schlüsselkompetenzen:

- der Kompetenz, selbstständig erfolgreich handeln zu können,
- der Kompetenz, mit Instrumenten der Kommunikation und des Wissens erfolgreich umgehen zu können und
- der Kompetenz, in sozial heterogenen Gruppen erfolgreich handeln zu können.

Darunter verstehen wir

- die Stärkung des selbstständigen Arbeitens zwischen den Polen „Eigeninteresse“ und „Anforderungen“,
- die Stärkung der Verantwortungsübernahme für das eigene Lernen,
- die Zielorientierung unseres Unterrichts (von Problemen zu Produkten) mit größtmöglicher Freiheit in den Arbeitsphasen (Raum für Experimente und Fehler),

- die Förderung des Gemeinschaftsgefühls durch Stärkung des Einzelnen in der Gruppe und in der ganzen Schule,
- dass alle am Schulleben Beteiligten ihre besonderen Stärken und Fähigkeiten zur Geltung bringen und damit Anerkennung finden können,
- unsere Schule als Ort gelebter Demokratie auf der Basis unserer „Schulcharta“ und
- Ruhe, Zeit und Gelassenheit im schulischen Miteinander.

Die Qualität unserer Arbeit in Unterricht und Schulleben misst sich für uns daran, dass sie auf die Nachhaltigkeit des Lernens setzt, auf die ständige Ergänzung und Erneuerung der beschriebenen Kompetenzen orientiert ist und schulisches Lernen eng zum außerschulischen Leben und Lernen in Beziehung setzt und verschiedene Lernorte nutzt. Wir arbeiten in unserer Schulgemeinschaft an der Entwicklung und Förderung der Gestaltungskompetenz in dem Wissen, dass die oben ausgewiesenen Kompetenzen nur an konkreten Inhalten, in Lernsituationen, die sich durch Methodenvielfalt auszeichnen und durch eine bewusste Reflexion und Bewertung des eigenen Lernens erworben werden können.

Die Weiterentwicklung unseres Konzeptes zur Entwicklung und Förderung der Gestaltungskompetenz unserer Schülerinnen und Schüler bedarf einer besonderen Begleitung. Die Entwicklung, Umsetzung und Evaluation soll dabei fortlaufend durch interne und externe Rückmeldeverfahren, die alle Mitglieder der Schulgemeinschaft einbeziehen, gestützt und abgesichert werden. Ziel ist ein kontinuierlicher Verbesserungsprozess in Unterricht und Schulleben, der erfahrbar und belegbar ist.

## **Das Durchführungsvorhaben**

In einer ersten dreimonatigen Konzeptphase ist ein mobiles Roboter-Labors am Luisen-Gymnasium entstanden, inkl. Einrichtung der Software, Bestellung der Hardware. Das didaktisch-pädagogische Konzept für die Arbeit mit Robotern im Informatikunterricht ist erstellt, sowie die beteiligten Lehrkräfte des Luisen-Gymnasiums durch einen vollständigen Kursus ausgebildet.

In den folgenden nächsten sechs Monaten wurden die durch das Luisen-Gymnasium, begleitet durch die Informatik der HAW, entwickelten Unterrichtseinheiten umgesetzt. Dabei wurde unterrichtsbegleitendes Material durch das Luisen-Gymnasium und die HAW Hamburg erstellt. Dieses Unterrichtsmaterial wird im weiteren Verlauf ständig erweitert. Das Kurt - Körber - Gymnasium hat seit März 2007 mit dem Einsatz von Robotern im Informatikunterricht begonnen, wobei dieser einem weiteren Unterrichtskonzept folgt. Dieser weitere Ansatz ist für eine Gegenüberstellung der Lernerfolge wichtig. Die didaktische Evaluation wird durch

---

die Informatik der HAW Hamburg in Kooperation mit der Fakultät für Erziehungswissenschaft, Psychologie und Bewegungswissenschaft der Universität Hamburg durchgeführt.

## 4 Vorbereitung

In der Vorbereitungsphase der Unterrichtseinheit sind verschiedenen Faktoren von besonderer Bedeutung. Zum einen müssen die Lernbedingungen gut durchdacht sein und zum anderen die technischen Begebenheiten. Die Lernumgebung umfasst unterschiedliche Teilbereiche, die sich von der direkten räumlichen Umgebung über mögliche Hilfsmittel zum direkten Umgang mit dem Computer und der Programmierung ziehen.

Bevor die Lernumgebung gestaltet werden kann, müssen die Rahmenbedingungen geklärt werden. Insbesondere muss hier auf die Nachhaltigkeit eines solchen Labores geachtet werden, z.B. Verbrauchsmaterialien.

### Räumlichkeiten

Der Klassenraum wäre im optimalen Fall gross genug, um die Tische im Halbkreis oder U-Form an den Seiten des Raumes aufzustellen, damit Hilfestellungen schnell und bequem gegeben werden können. Der Lernprozess und die Arbeit an den Computern kann so von allen Seiten gut beobachtet werden. Arbeitet man, wie es in diesem Projekt der Fall ist, mit Robotern, so wird in der Mitte des Raumes eine weitere grössere Arbeitsfläche benötigt, auf der Programme mit dem Roboter getestet werden können. Doch nicht nur als Testfläche bietet diese zentrale Arbeitsfläche Vorteile, die Arbeit im Plenum wird so persönlicher und intensiver gestaltet, Erläuterungen oder Präsentationen des Lehrer/-Innen werden besser von der Gruppe verfolgt. (Guerin, 2006)

In den meisten Fällen lassen bauliche oder allgemein schulische Bedingungen eine optimale Gestaltung des Klassenraumes und der Lernumgebung nicht zu. In derzeitigen Unterricht am Luisengymnasium dient der Physikraum als Arbeitsraum, die Tische sind demnach in Reihen aufgestellt. Die Testfläche für die Roboter wird vor der Tafel, am Lehrertisch stattfinden und auf einer eigens dafür gebauten Testfläche. Diese Testfläche wurde nach den Normen der Lego Rescue League gestaltet. Hier steckt die Idee eine zusätzliche Motivation zu schaffen, indem den Schülern als Ziel eine Teilnahme an diesem internationalen Wettbewerb vorgeschlagen wird.

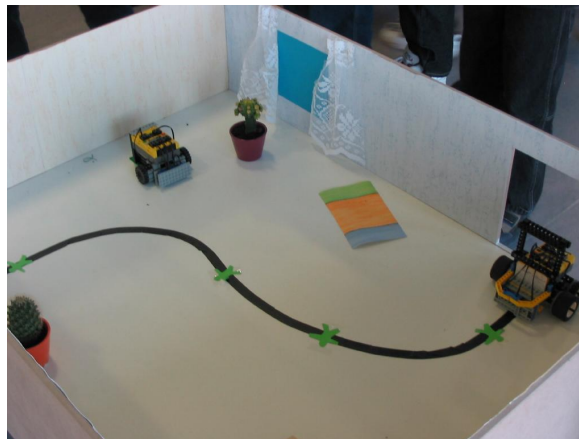


Abbildung 4.1: Rescue League

## Zeitplanung

Bevor der Unterricht beginnen kann, ist es sinnvoll eine grobe Zeiteinteilung mit Teilzielen zu formulieren, damit eine Struktur im Verlauf der verschiedenen Unterrichtsphasen erkennbar bleibt. Nach Beginn des Unterrichts besteht die Möglichkeit, dass der ursprüngliche Leitfaden ausser Acht gerät, da durch Zwischenfragen und Anregungen ein anderer Schwerpunkt im Unterricht entstehen kann, als ursprünglich gedacht. Grundsätzlich ist neuen Ideen und Entwicklungen viel Raum zu gewähren, dennoch kann der Kern der Unterrichtseinheit nicht vergessen werden. Dieser Kern bildet sich aus den Ansprüchen des Lehrplans an die jeweilige Klassenstufe und den allgemeinen Vorkenntnissen der SchülerInnen.

In dieser Form des Unterrichts nimmt eigenständiges Lernen einen weiten Raum ein, damit entstehen neue Ideen und neue Problemlösungsansätze, die in das bestehende Unterrichtskonzept eingebettet und reflektiert werden müssen. Reflektion ist ein wesentlicher Teil des Unterrichtes, da so die Kritikfähigkeit der SchülerInnen geschult und der Unterricht transparent wird. Transparenz des Unterrichtes ist wichtig, kennen die SchülerInnen die Aufgabenstellung, so können sie nicht nur eigene Vorstellungen hinzufügen, sondern reflektieren ihre eigene Arbeit im Hinblick darauf. Dazu muss eingangs die Aufgabe deutlich vorgestellt und besprochen werden, zum Teil kann es erforderlich sein Theorieeinheiten vor die Arbeit an der Simulation zu setzen, damit später der Arbeitsfluss nicht weiter gestört wird. Weitere Erklärungen und Theorieeinheiten müssen nach Bedarf in den Unterricht eingestreut werden. Wie bereits angedeutet, ist es wichtig sich einen Zeitplan oder Etappen zurechtzulegen, in dem verschiedene Teilziele benannt werden. So behält man selbst einen Überblick über den Arbeitsaufwand und kann die Aufgabenstellungen dementsprechend variieren. Zu jedem Zeitpunkt sollten unterschiedliche Schwierigkeitsgrade in der Aufgabenstellung möglich sein, um schneller arbeitenden Gruppen neue Aufgaben präsentieren zu können, falls diese nicht eigenständig auf neue Ideen stossen. Die Etappen sollten für die Klasse offen gelegt



werden, damit sie sich an einem Ablauf orientieren können und gegebenenfalls eigene Vorschläge einbringen können. Diese Offenheit des Unterricht ist eine neue Herausforderung für Lehrkräfte und kann ungewohnte Kritik von den SchülerInnen mit sich bringen auf die im Unterricht eingegangen werden muss. Der Lehrer steht damit nicht mehr im Zentrum des Unterrichts, sondern gibt mehr Freiraum für Diskussionen und eigene Erfahrungen der SchülerInnen, die er als Lernbegleiter möglichst indirekt lenkt.

## Hilfestellung

Es gibt diverse Hilfsmittel in denen die SchülerInnen selbst nach Problemlösungen suchen können. So könnte man zum Beispiel Internetseiten oder Bücher (Hans-Dieter Burkhardt, 2006), (Norbert Breier, 2003) als Quellen empfehlen. In diesem Schulprojekt soll für die SchülerInnen ein Nachschlagewerk entstehen, in dem sie die wesentlichen Grundbegriffe und Definitionen nachlesen können. Optimal wäre eine gemeinsame Ausarbeitung der Begriffe, da so die SchülerInnen in ihren eigenen Worten die Bedeutungen festhalten und verstehen. Zur Überprüfung des allgemeinen Verständnisses und zur späteren Notenvergabe werden gruppenweise Protokolle erstellt, wobei jeweils eine Gruppe pro Sitzung ihr Protokoll ins Schulnetz und somit den anderen zur Verfügung stellt. Hintergrund dieser Arbeit ist neben Kontrolle und Notenvergabe, dass ein allgemeines Informationsnetz für SchülerInnen entsteht, auf das die momentane und spätere Arbeitsgruppen zugreifen können. Ein wesentliches Hilfsmittel der SchülerInnen ist der Austausch untereinander, die SchülerInnen unterstützen sich gegenseitig bei Problemen, bzw. geben Hilfestellungen, wenn sie ein Problem vor den anderen gelöst haben. Dadurch wird die Rolle des Lernvermittlers vom Lehrer genommen und ins Plenum getragen, so wird das eigenständige Lernen gefördert und das Selbstvertrauen der SchülerInnen gestärkt. Für den Lehrer/-Innen ist diese Form des Unterrichts und des gemeinsamen Lernens eine neue Rolle, dies kann unter Umständen zu Problemen im Selbstverständnis des Lehrers führen. Im klassischen Unterricht führt der Lehrer den Unterricht und übernimmt die Rolle des Wissensvermittlers, hier gibt er diese aus der Hand und gesteht damit eigene Unsicherheiten ein. Die Rolle als Beobachter und Lernbegleiter ist neu in der Schule und weicht von vielen bestehenden Unterrichtsformen, nach denen SchülerInnen reproduzieren, aber nicht reflektieren, ab. SchülerInnen sind meist sehr flexibel in der Gestaltung des Unterrichtes, während Lehrkräfte sich in diesem Bereich des Projektes zurücknehmen und neue Ideen zulassen müssen.

## Unterricht:

Nachdem die allgemeinen Vorbereitungen abgeschlossen sind, schliesst sich die Planung des Unterrichtes an. Viele Anregungen zu diesem Thema lassen sich bei Matthias Stolt

„Roboter im Unterricht“ (Stolt, 2001) finden. Hierzu muss das Themenfeld in unterschiedliche Abschnitte oder Einheiten unterteilt werden, damit der gedankliche Rote Faden im Laufe des Projektes nicht in den Hintergrund gerät. Es ist wichtig den Erfahrungsstand der SchülerInnen zu kennen, um die Einteilung des Unterrichtes darauf abzustimmen. Die Hilfsmittel, die wie bereits zuvor beschrieben für die eigenständige Recherche der SchülerInnen wichtig sind, sollten auf diesen Erfahrungshintergrund abgestimmt sein.

Die Vorbereitungen des Unterrichts belaufen sich nicht nur auf die äusserlichen Begebenheiten, das Lernmaterial oder die Einteilung der Gruppen, der Lehrer/-Innen selbst muss sich auf den Unterricht und das Thema einstellen. Es können nicht nur fachliche Herausforderungen an ihn gestellt werden, auch das eigene Rollenverständnis kann sich verändern. Indem SchülerInnen an eigenverantwortliches und selbstständiges Lernen herangeführt werden, verändert sich die Rolle des Lehrenden in die des Lernbegleiters, eine durchaus ungewohnte Position im Schulalltag. Wie weit die Selbstständigkeit der Klasse geschult werden soll oder kann, hängt mit der Klassenstufe und dem Vorwissen der Klasse zusammen.

Die bereits erwähnten Teilziele müssen vor Beginn des Unterrichts feststehen, damit ein allgemeines Gerüst für den Unterricht steht. Diese Ziele werden sich im Laufe des Unterrichtes etwas wandeln, da sich erst dann der eigentliche Wissenstand der Schüler/-Innen nachvollziehen lässt und mögliche Theorieeinheiten und andere Erklärungsmuster herangezogen werden können. Es besteht die Möglichkeit nicht mit exakten Teilzielen für jede Unterrichtsstunde zu arbeiten, sondern sich „Meilensteine“ zu setzen, in denen festgehalten wird wann welches Ziel oder welche Erfahrung sinnvoll gemacht sein sollte. Diese Meilensteine und Teilziele helfen der Lehrkraft eigene Unsicherheiten zu kontrollieren und sich im Unterricht nicht in (derzeit) unwesentlichen Elementen zu verlieren.

Die Einführungssequenz ist sehr wichtig. Sie stellt die Hauptmotivation dar. In dieser Sequenz wird ein Überblick über bestehende Projekte oder Entwicklungen gegeben, ein Ausblick auf Möglichkeiten in der Schule oder in Wettbewerben gezeigt und die Aufgabenstellung vorgestellt. Hier sollten viele unterschiedliche Reize /Medien benutzt werden, um den Vortrag zu unterstreichen.

## 5 Technik

Die technischen Begebenheiten lassen sich wie folgt beschreiben: Der zu programmierende Roboter ist der LEGO NXT, eine Weiterentwicklung der LEGO Mindstorms. Das Herzstück des NXT ist der NXT-Stein, in dem ein 32-bit-Prozessor verbaut ist. Der NXT verfügt über 3 Servo-Motoren und jeweils einen Ton-, Ultraschall-, Licht- und Tastsensor. Über Bluetooth oder USB 2.0 kann der NXT an einen PC oder Mac angeschlossen werden. Die Programmiersprache und die Entwicklungsumgebung LEGO hat für den NXT eine eigene Programmiersprache entwickelt. Hierbei handelt es sich um eine auf Symbolen beruhende grafische Programmiersprache, bei der per Drag-and-Drop die Programme zusammengestellt werden.

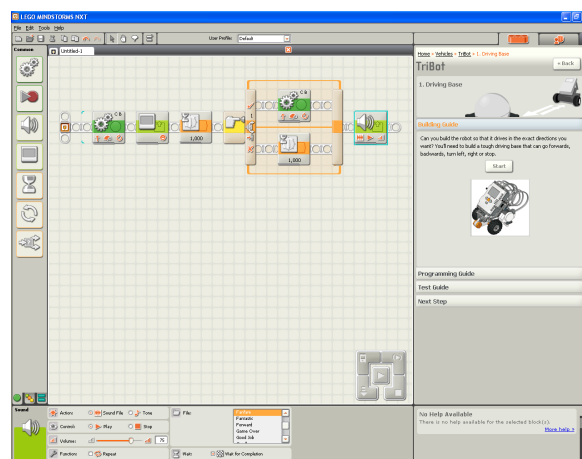


Abbildung 5.1: Legoumgebung

Diese einfache, sehr intuitiv zu benutzende Sprache besitzt leider nicht die Mächtigkeit einer „richtigen“ Programmiersprache. So sind zwar einige Konzepte der Informatik implementiert, wie z.B. Schleifen. Andere fehlen jedoch komplett oder sind nicht offensichtlich als solche erkennbar. Außerdem besteht durch den simplen Aufbau des Programms eine und die fehlende Tiefe, die Gefahr der Unterforderung der SchülerInnen.

Deshalb wurde als Programmiersprache C# und als Entwicklungsumgebung das Microsoft Visual Studio mit dem Microsoft Robotics Studio gewählt. (Gunnerson, 2000) C# ist eine objektorientierte Programmiersprache, die im professionellen Sektor weit verbreitet ist. Ihre

Syntax ist der von anderen objektorientierten Sprachen, z.B. Java, sehr ähnlich, wodurch ein Umstieg auf solche Programmiersprachen in höheren Jahrgangsstufen leicht möglich ist. Außerdem können so vorhandene Vorkenntnisse der Lehrenden optimal genutzt werden.

Das Microsoft Robotics Studio ist ein „Software Development Kit“ für die Entwicklung von Roboter-Programmen. Es stellt eine Sammlung von Diensten und Klassen bereit, mit denen man eine große Anzahl von unterschiedlichen Roboter-Plattformen programmieren kann. Unterstützt werden bisher Roboter von Fischer Technik, Pioneer und LEGO. Das Robotics Studio abstrahiert vollständig von der anzusprechenden Roboter-Hardware. So ist es ohne Probleme möglich ein Programm, das einen Pioneer ansteuert für einen LEGO NXT zu benutzen.

Durch den integrierten Simulator ist es möglich auch ohne Roboter Programme zu entwickeln und zu testen. Der Simulator ist eine physikalisch korrekt berechnete 3D-Umgebung, die mit DirectX programmiert und leicht anpassbar ist.

Die Simulation dient dazu den SchülerInnen die Arbeit mit dem Roboter und der Programmierung näher zu bringen und den Einstieg zu erleichtern. Dabei wird die Simulation wie ein Computerspiel gestaltet, eine Umgebung in der sich die meisten Teilnehmer zurechtfinden.

Diese Idee des Einstiegs dient nicht nur dazu die ersten Ängste vor einem kompliziert anmutendem Programm zu nehmen, sondern weckt Interesse an Programmierung, indem die zu benutzenden Befehle auf ein Minimum reduziert und die Programmierumgebung einfach gehalten wird. Dafür wurden komplizierte Strukturen so „versteckt“, dass sie bei Bedarf später aufgedeckt und erklärt werden können. Zunächst hilft diese Einfachheit der zu benutzenden Befehle Begriffe einzuführen und diese anhand des Codes zu erklären. Das Programm und der benutzte Code wird so für die SchülerInnen plastisch. Sie können ihre Befehle direkt in der Simulation testen und die eventuell erhaltenen Fehlermeldungen lösen. Die Einführung in das Thema Robotik durch die Simulation hat weitere Vorteile. Die Klasse konzentriert sich zunächst auf die Programmierung und nicht auf die Gestaltung des Roboters. Es wird nur mit dem Notebook gearbeitet, so liegt das Hauptmerk auf einer fehlerfreien Lösung der Aufgabe. Die Arbeit mit einem realen Roboter führt vielfältige Probleme mit sich, die bei der Konstruktion und der Programmierung bedacht werden müssen. Zwar hilft der „Spieltrieb“ bei der Lösung sehr, doch stellt sich durch die Vielzahl Ursachenmöglichkeiten schnell Frustration ein. Durch den Simulator ist es möglich sich zunächst vollständig auf die Lösung der Informatik-Probleme zu konzentrieren, ohne durch andere Probleme, z.B. durch die Roboter-Hardware oder Probleme der Umgebung gestört zu werden.

Das Robotics Studio unterstützt die von Microsoft im .net-Framework angebotenen Programmiersprachen, wie z.B. Visual Basic oder C#. Aufgrund der genannten Vorteile wurde sich für C# entschieden.

C# ist eine professionell eingesetzte objektorientierte Programmiersprache, deren Nutzung

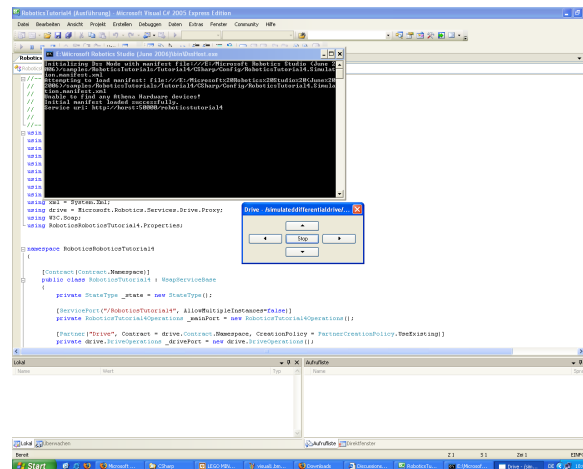


Abbildung 5.2: Entwicklungsumgebung C#

sowohl in der Sekundarstufe I als auch in der Sekundarstufe II möglich ist. Hierfür muss jedoch eine altersgerechte Anpassung der Entwicklungsumgebung im weitesten Sinne erfolgen. Je älter die Schülerinnen und Schüler sind desto anspruchsvoller kann die Entwicklungsumgebung sein. Gleiches gilt für die Aufgabenstellungen. Die Workbench muss also einfach gestaltet sein, um in unteren Klassen mit ihr arbeiten zu können, ohne die Schülerinnen und Schüler zu überfordern. Sie muss aber gleichzeitig erweiterbar sein, damit auch höhere Klasse mit ihr arbeiten können und nicht unterfordert werden.

## Entwurf einer Workbench für ein Robot-Building-Lab an Schulen

Um eine einfache Benutzbarkeit herzustellen wird den Schülerinnen und Schülern nur ein begrenzter Befehlssatz zur Verfügung gestellt. Der Befehlssatz beschränkt sich auf die Ansteuerung der Motoren und die Statusabfrage bestimmter Sensoren. Durch die Einführung einer 'Robot-Klasse' kann die Komplexität des Microsoft Robotics Studio gekapselt werden. Die Schülerinnen und Schüler arbeiten nur in der „Robot- Klasse“, die genaue Funktionsweise bleibt ihnen verborgen.

Dies ist sehr wichtig, um eine Überforderung der Schülerinnen und Schüler zu vermeiden und sie nicht durch zu hohe Anforderungen schnell zu demotivieren. Die „Robot-Klasse“ selbst ist ebenfalls sehr einfach aufgebaut, so dass die Schülerinnen und Schüler sie leichter nachvollziehen und modifizieren können. Die Vereinfachungen der Sprachelemente sind nur sehr begrenzt vorgenommen worden. Der ursprüngliche Charakter von C# soll erhalten bleiben. Schülerinnen und Schüler sollen sich an die Sprache gewöhnen.

Trotz der Einfachheit ist der volle Sprachumfang von C# verfügbar und somit auch die grundlegenden Konstrukte der Programmierung, wie Variablen, Schleifen oder Verzweigungen.

Der volle Sprachumfang ermöglicht es in höheren Jahrgangstufen die Workbench als ganzes zu modifizieren. Die „Haube“ kann aufgeklappt werden und die dahinterliegende Technik nachvollzogen werden. Eigene Modifikationen sind möglich und weitere Konzepte der Programmierung können genutzt werden, wie z.B. Threads oder Zustände.<sup>1</sup>

## Der Computer

Der Mangel an Platz in den Räumlichkeiten des Luisengymnasiums macht die Nutzung von Notebooks notwendig, da sonst entwickeln und Roboter bauen parallel nicht möglich ist. Die Notebooks müssen folgende Systemanforderungen erfüllen: Prozessor mit 600 MHz oder schneller empfohlen, Betriebssystem: Windows Server 2003; Windows XP Professional Service Pack 2, Arbeitsspeicher: mind. 256Mb (Visual Express), 512Mb sollten es für flüssiges Arbeiten schon sein, freier Festplattenspeicher: 1,5 GB, Grafikkarte: z.B. ATI RADEON X1300 oder NVIDIA Quadro NVS 110 und Bluetooth. Die von Dell gelieferten Notebooks erfüllen die o.g. Anforderungen, vor allem die Grafikkarte ist für die Nutzung der Simulation sehr wichtig.

---

<sup>1</sup>(Rosseburg, 2007)

## 6 Der Unterricht am Luisengymnasium



Abbildung 6.1: Lego NXT - konstruiert von Schülern

Der Unterricht im Wahlpflichtfach „Roboter“ setzt sich aus zwei Einheiten zusammen, zum einen der Arbeit mit einer Simulation und der Arbeit mit dem Lego Mindstorm NXT. Der erste Teil dieser Einheit wurde vor Weihnachten abgeschlossen. In diesem Bereich konnten die Schülerinnen und Schüler erste Erfahrungen im Umgang mit Programmierung und Umsetzung des erstellten Programms sammeln. Dabei bleibt zu berücksichtigen, dass diese Klasse sehr geringe bis keine Grundkenntnisse in der Informatik besitzt. In dieser Unterrichtseinheit haben die Schülerinnen und Schüler einen ersten Einblick in Programmiertechniken erhalten und in die typischen Problemstellungen, die sich aus der Arbeit mit der Simulation ergeben. Die Simulation dient dazu den SchülerInnen die Arbeit mit dem Roboter und der Programmierung näher zu bringen und den Einstieg zu erleichtern. Dabei wird die Simulation wie ein Computerspiel gestaltet, eine Umgebung in der sich die meisten Teilnehmer zurechtfinden. Diese Idee des Einstiegs dient nicht nur dazu die ersten Ängste vor einem kompliziert anmutendem Programm zu nehmen, sondern weckt Interesse an Programmierung, indem die zu benutzenden Befehle auf ein Minimum reduziert und die Programmierumgebung einfach gehalten wird. Dafür wurden komplizierte Strukturen so „versteckt“, dass sie bei Bedarf später aufgedeckt und erklärt werden können. Zunächst hilft diese Einfachheit der zu benutzenden Befehle Begriffe einzuführen und diese anhand des Codes zu erklären. Das Programm und der benutzte Code wird so für die Schülerinnen und Schüler plastisch. Sie können ihre Befehle direkt in der Simulation testen und die eventuell erhaltenen Fehlermeldungen lösen.

Die Einführung in das Thema Robotik durch die Simulation hat einen weiteren Vorteil: Die Klasse konzentriert sich zunächst auf die Programmierung und nicht auf die Gestaltung des Roboters.

Die erste Einheit des Unterrichtes bestand in einer theoretischen Einführung in Informatik und Roboter. Zunächst gab es eine Übersicht über allgemeine Informationen zum Fachbereich Informatik und Roboter, indem Roboterarten, Projekte und Möglichkeiten des Einsatzes von Robotern in einem Vortrag dargestellt wurden. Anschliessend stellten die beiden teilnehmenden Informatikstudenten die Simulation, die Programmiersprache und die Aufgabenstellung vor.

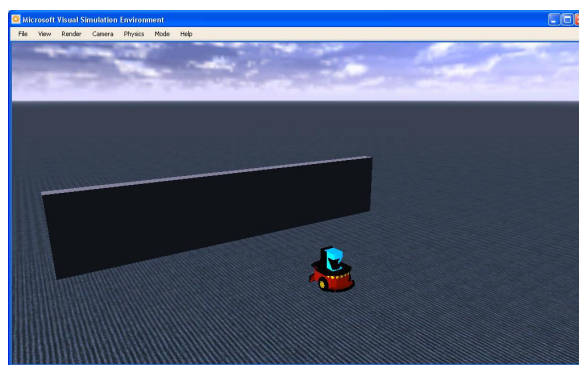


Abbildung 6.2: Robotersimulation

Nachdem die SchülerInnen diese Aufgabe und den Ausblick auf die Arbeit mit dem Lego Mindstorm NXT erhalten haben, wurden die Notebooks präsentiert, die jede Gruppe erhält. Die Gruppe ist dabei für ihr Notebook verantwortlich. Am Ende der Unterrichtseinheit sind die ersten Fragen geklärt und die Gruppenbildung abgeschlossen. Zu Beginn der zweiten Unterrichtseinheit sollten die SchülerInnen versuchen ein Hindernis innerhalb der konstruierten Welt mit einem Roboter einwandfrei zu umrunden. Als Hilfsmittel standen ihnen „Laser“ und „Bumper“ zur Verfügung. Diese Aufgabe war eine erste Vorübung für die „Welt“ der Robocup Junior Rescue League<sup>1</sup>, in der später mit dem Lego-Mindstorm NXT gearbeitet werden könnte. Die Begriffe der Aufgabenstellung wurden geklärt und mögliche Probleme aufgezeigt. Die Bearbeitung der Aufgaben wurde immer wieder individuell oder allgemein unterbrochen, um Probleme oder weitere Arbeitsschritte zu erklären, wobei darauf geachtet wurde, dass die SchülerInnen einen möglichst grossen Anteil an Lösungsvorschlägen einbringen. Die Lernbegleiter stellten keine Lösungen ohne Mitwirkung der SchülerInnen auf, um den SchülerInnen die Möglichkeit zu geben sich allein in die Aufgabenstruktur einzuarbeiten. Damit wurde ihre Aufmerksamkeit für Programmierung und die dabei auftretenden Probleme geschult und sie wurden unbemerkt auf neue Betrachtungsperspektiven hingeführt, die bei späteren Aufgaben nützlich sein könnte. Sie erkannten, dass die abstrakte Welt

<sup>1</sup> (fraunhofer:2006, 2006)



der Simulation natürlichen Gesetzen unterlag und sie auf reale Probleme Rücksicht nehmen müssen. Am Ende der Doppelstunde hatten die Gruppen die Aufgabenstellung gelöst. Die dritte Unterrichtseinheit wurde als Theorieeinheit abgehalten und beschäftigte sich mit Begrifflichkeiten und weiterführenden Erklärungen. Teile des erstellten Programmcodes wurden mit Hilfe eines Beamers an die Wand reflektiert und im Plenum besprochen. Die Notebooks blieben während dieser Phase des Unterrichts geschlossen, damit keine Ablenkungen oder Unruhe entstehen konnte. Bislang hatten die SchülerInnen weitgehend Programmteile reproduziert, um einen ersten Eindruck von Programmierung und eine Sicherheit in dem Umgang mit dem Programm zu bekommen. Jetzt wurden Teile des bisherigen Codes zerlegt und erklärt. Innerhalb dieser Erklärungen konnten Elemente der neuen Aufgabenstellung mit den SchülerInnen zusammen erarbeitet werden, so dass eigene Ideen und Fragen bearbeitet werden konnten. Im zweiten Teil dieser Einheit setzten die Gruppen die erweiterte Aufgabenstellung um, diesmal nicht nur reproduzierend, sondern sie suchten nach eigenen Möglichkeiten innerhalb des Codes, um die Aufgabe zu lösen. Dabei standen ihnen wieder der Lehrer und die Studenten als Hilfsquellen zur Verfügung, wobei es wichtig war, dass sie nur begleitend arbeiteten. In der folgenden vierten Unterrichtseinheit wiederholte die Klasse Theorieelemente, um das gewonnene Wissen zu festigen. Die Aufgabenstellung wurde noch einmal umrissen und Fragen der SchülerInnen geklärt. Der Grossteil der Stunde bestand in der Arbeit an der Simulation und der Lösung der Aufgabenstellung. Die dabei entstehenden Fragen wurden individuell oder im Plenum geklärt. Die SchülerInnen unterstützen sich gegenseitig bei Fragestellungen und gaben sich Tipps, wie man die Aufgabe besser gestalten oder lösen konnte. Das Interesse der SchülerInnen bestand nicht allein in der Lösung ihrer eigenen Aufgaben, sondern entwickelte sich weiter zur Gestaltung von neuen Aufgaben, Fragen an die Simulation und der freiwilligen Hilfestellung bei anderen Gruppen. Diese Hilfestellung untereinander bietet mehr Vorteile für die SchülerInnen, als die Hilfestellung durch die Lehrkraft oder Studenten, da sie ihre Ideen miteinander diskutieren und eigene Vorschläge genauer betrachten. Werden eigene Gedanken mit der Lehrkraft besprochen, kann es schnell dazu kommen, dass die Argumente des Lehrenden als „richtige“ Lösung angesehen werden und eigene Ideen verworfen werden. Gründe hierfür liegen zum einen in der gewohnten Unterrichtsform und dem SchülerInnen -LehrerIn -Verhältnis und zum anderen in dem Erfahrungsschatz, der der Lehrkraft von Seiten der SchülerInnen zugewiesen wird. Am Ende dieser Unterrichtseinheit konnten viele Ideen der SchülerInnen in der Simulation umgesetzt werden. Die fünfte Unterrichtseinheit beschäftigte sich allein mit der Arbeit an der Simulation. Die SchülerInnen ergänzten die vorherige Aufgabenstellung selbst und probierten neue Programmierungselemente aus. Bei Problemen oder Fragen wurde sich Hilfe durch den Lehrer, Studenten oder auch anderen Schülergruppen organisiert. Auffällig an dieser Einheit war, dass sich die einzelnen Gruppen konzentriert um ihre Aufgabe kümmerten, obwohl es die fünfte Doppelstunde war in der sie innerhalb der Simulation arbeiteten. Bei der Planung der Stunden kam die Befürchtung auf, es könne den SchülerInnen nach dieser Zeit etwas „langweilig“ werden, wenn sie wieder an der Simulation arbeiten würden,

doch das Gegenteil traf ein. Zum Teil wird dies damit zusammenhängen, dass die Simulation einen Computerspiel nachempfunden ist und damit ein vertrautes Umfeld mit neuen Möglichkeiten darstellt. Während der sechsten Einheit in der Arbeit mit der Simulation wurde das Nassi-Shneiderman Diagramm behandelt. Nach der Zeit der praktischen Arbeit, sollte eine Theorieeinheit neue Elemente einführen und festigen. Das Nassi-Shneiderman Diagramm war nützlich, um das Programmieren auf eine neue Ebene zu setzen und gleichzeitig Fragen zu klären. Zunächst wurde das Diagramm gemeinsam besprochen und zusammen mit Hilfe des Diagramms ein Programm erstellt. Hiernach konnte die Klasse ihr eigenes Programm in die Strukturen des Nassi-Shneiderman Diagramms einfügen und neue Ideen zunächst auf dem Papier programmieren. Für die Klasse war dies eine neue Erfahrung, der nicht alle gegenüber aufgeschlossen waren. Die Arbeit mit der Simulation hat einen starken spielerischen Charakter und ist damit viel reizvoller, als die Programmierung auf dem Papier. Es ist jedoch wichtig, dass sie auch diese Erfahrung machen, damit weitere Elemente der Informatik eingeführt werden können und in späteren Klassen auf dieses Basiswissen zurückgegriffen werden kann. Zudem ist es anhand des Nassi-Shneiderman Diagrammes möglich ohne Computer zu Programmieren. Ein weiterer Grund für die Einführung des Diagrammes war, daß viele Teile in der Programmierung unklar für die SchülerInnen waren. Die Arbeit mit dem Diagramm sollte eine Hilfestellung im Umgang mit Programmierung und im Verständnis mit ihr geben. Es wurden Teile des selbsterstellten Programms der SchülerInnen in das Diagramm eingesetzt, damit die SchülerInnen verstehen, warum ein Programm wie geschrieben wird und es deutlich dargestellt wird. Die deutliche Darstellung eines Programmcodes ist wichtig, damit Fehler besser gefunden werden können und damit jemand anderes später den Code nachvollziehen kann. Dies ist ein wichtiges Element der Informatik wie auch in anderen Unterrichtsfächern.

Die SchülerInnen erstellten zu jeder Einheit Protokolle, die als Lernerfolgskontrolle galten und die spätere Note der SchülerInnen bildeten. Anhand dieser Protokolle wurden verschiedene Verständnisprobleme deutlich, auf die in weiteren Unterrichtseinheiten aufgebaut werden konnte.

In der letzten Doppelstunde vor den Weihnachtsferien wurde mit dem Zusammenbau der Lego Mindstorm NXT begonnen. Die SchülerInnen erhielten den Auftrag die Roboter eigenständig, d.h. nicht an die beiliegende Anleitung, sondern sich an eigene Ideen zu halten. Das wesentliche dabei war, dass die Roboter flach gebaut werden mussten, um über eine Rampe in Probemodell zu gelangen. Die Umsetzung der eigenen Ideen war wichtig, damit sich die Klasse Gedanken um die Nützlichkeit, Technik, Statik und Design des Roboters macht. Dabei muss bedacht werden, dass nicht alle SchülerInnen die gleichen Vorkenntnisse im Umgang mit Lego besitzen, dementsprechend kann das Zusammenbauen der Roboter ohne allgemeingültige Anweisung länger dauern. Die Phase des Zusammenbauens der Roboter hielt nach den Ferien an. In den Ferien hatten sich viele der SchülerInnen weitere Gedanken um den Zusammenbau der Roboter gemacht, teilweise gab es Zeichnungen, wie der Roboter

besser gebaut werden könnte und welche Vorteile welcher Aufbau bieten würde. Besonders auffällig war eine Schülergruppe, die sich in der Ferien zu den Modellgedanken eine Programmierung überlegt hatten. Dabei hatten sie das Programm auf drei DIN A4 Seiten geschrieben, dies verdeutlicht, dass die Nassi-Shneiderman Stunden für viele SchülerInnen nützlich waren, um eigenen Ideen mit wenigen Mitteln umzusetzen. Die Gruppe erklärte ihre Programmierung, ohne die Hilfsmittel Computer oder Roboter, der sich noch im Aufbau befand, allerdings mit Tafelschwamm und Hindernissaufbau aus Tafelkreide auf dem Lehrerpult. Dieses Beispiel macht deutlich, wie sehr dieses Thema die SchülerInnen beschäftigt und sie so spielerisch Grundlagen erlernen können. Der Bau der Roboter dauerte noch weitere 2 Doppelstunden an, wobei bereits erste Schritte und Testläufe mit den Robotern stattfinden konnten.

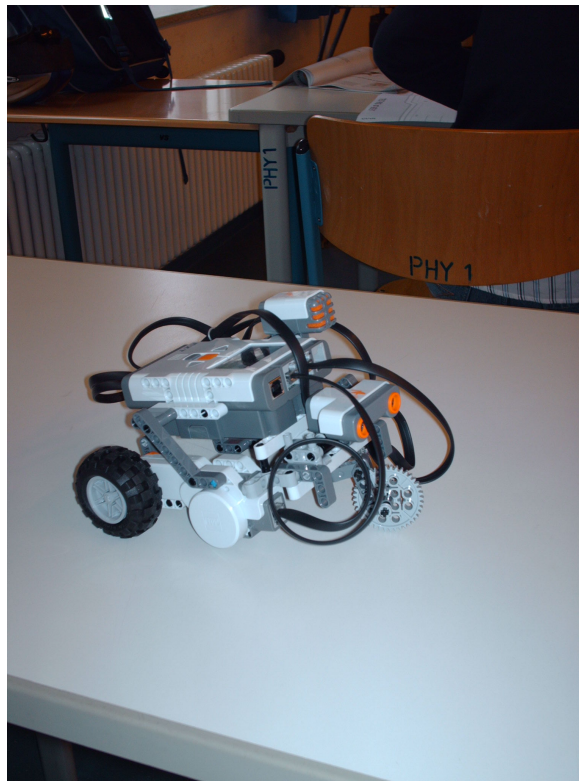


Abbildung 6.3: weiterer NXT, konstruiert von SchülerInnen

Die ersten Testläufe der Roboter wurden mit der Software auf dem Lego NXT abgehalten. Das Programm beinhaltet einfaches Vor- und Rückwärtsfahren, kann jedoch um die Motoren und den ersten Aufbau des Roboters zu testen sinnvoll sein. Nachdem die Roboter fertiggestellt waren, erhielten die SchülerInnen eine Erklärung wie sie die gelernten Programmierelemente auf den Roboter übertragen können. Die Klasse erarbeitete an Aufgaben, die zum Teil selbstgestellt und zum Teil von der Lehrkraft vorgegeben wurden. Dabei

wurden immer wieder selbstständig neue Lösungswege gesucht, wobei nach dem Versuch und Irrtum Prinzip vorgegangen wurde. Die SchülerInnen wurden in der Arbeitszeit nicht müde sich mit dem Themenbereich Roboter auseinanderzusetzen. Durch das grosse Interesse an den Robotern lernten die SchülerInnen unbemerkt viele Grundlagen der Informatik und setzten sich mit Technikproblemen und Designfragen auseinander.

Die Schülerinnen und Schüler arbeiten weitgehend eigenständig, d.h. sie versuchen selbstständig typische Probleme im Programmieren zu lösen, wobei sie ständig Hilfestellungen und Begleitung durch den Lehrer oder die anwesenden Studenten bekommen können. Ergeben sich komplexe Fragestellungen werden diese im Plenum angesprochen. Allen Mitwirkenden steht ein Chatroom innerhalb des Luisennetzes zur Verfügung über den sie zum einen ihre Erfahrungen und Protokolle veröffentlichen können, zum anderen können sie dort Hilfestellungen bei Problemen erhalten. Diese Arbeitsweise stärkt das eigenverantwortliche Handeln der Klasse und gibt ihnen Sicherheit im Umgang mit komplexen Aufgaben. Die gewonnenen Eindrücke und das Selbstbewusstsein im Umgang mit dem Computer und dem Programmiercode geben den Schülerinnen und Schülern in ähnlichen Situationen Sicherheit im Umgang und eventuell Lösungssuche von Problemen. Der Unterricht soll Muster für solche späteren Situationen geben. Der Rahmenplan<sup>2</sup> für das Wahlpflichtfach Informatik sieht diese Kompetenzentwicklung als einen wesentlichen Leitfaden für den Unterricht an. Der Informatikunterricht soll den Schülerinnen und Schülern ein Orientierungswissen vermitteln, anhand dessen sie elektronisch verfügbare Informationen leichter erschliessen, strukturieren und erarbeiten können. Die Arbeit mit der Simulation führt die Klasse an dieses Wissen heran und lässt den Teilnehmern dabei genügend Raum für die Umsetzung eigener Ideen und Darstellungen. Im späteren Verlauf des Projektes, in der Arbeit mit realen Robotern, wird diese Flexibilität und Kreativität der Schülerinnen und Schüler sehr wichtig sein, um die entstehenden Probleme zu lösen.

Die Reaktion der Klasse auf das Thema Roboter war sehr gespannt. Ihre erste Neugier, was sie in diesem Kurs erwarten würde, wurde nach einer anfänglichen Scheu schnell in Interesse gewandelt. Jede Gruppe, die aus 2-3 Personen besteht, verfügt über ein eigenes Notebook auf das eigenverantwortlich geachtet werden muss. Die Gruppenarbeit wird, wie bereits erwähnt, möglichst selbstständig gehalten. Dabei helfen sich die Gruppen bei Fragestellungen gegenseitig und geben sich Tipps in der Lösungsgestaltung. Viele Gruppen haben mit ihrer Lösungsvariante überrascht, da mit einigen Ideen von Seiten der Lehrkräfte nicht gerechnet wurde. Der Ideenreichtum und die Einsatzbereitschaft von einigen Schülern ging so weit, dass sich zum Teil, wie schon erwähnt, in den Weihnachtsferien Programme für den realen Roboter in der nächsten Unterrichtseinheit überlegt und verschriftlicht wurden. Einige Schülerinnen haben mit extrem kurzen und prägnanten Lösungen für Begeisterung gesorgt.

---

<sup>2</sup>(Freie und Hansestadt Hamburg, 2003)



Abbildung 6.4: Schüler mit NXT

Die spielerische Atmosphäre in dem Projekt führt dazu, dass sich Ideen besser entfalten können. Der Lerncharakter wird nicht durch strenge Vorgabe des Unterrichtsmaterials bestimmt, sondern durch Anregungen der Schülerinnen und Schüler mitbestimmt. Die Klasse hatte vor Beginn des Projektes keine Erfahrung im Programmieren, die fehlenden Grundlagen müssen weiterhin in den Unterricht eingestreut werden, ohne dass der spielerische Charakter verloren wird, da die Klasse so unbewusst viel Wissen sammelt.

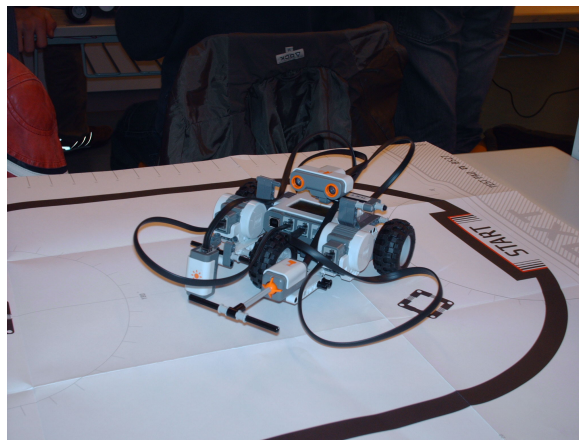


Abbildung 6.5: NXT konstruiert von SchulerInnen

Der Kompetenzerwerb der Schülerinnen und Schüler bezieht sich nicht nur auf den fachlichen, sondern auch auf den sozialen Bereich. Gruppenarbeit, wird sie selbstständig durchgeführt, vereint sehr viele Lernaspekte in sich, z.B. Planung der Vorgehensweise, Entscheidungen, Verteilung und Präzisierung von Aufgabenstellungen, die Einhaltung von Absprachen und die Kommunikation und das Vertrauen zwischen den Gruppenmitgliedern. Die unterschiedlichen Perspektiven, die sich in der Gruppenarbeit ergeben, müssen besprochen

werden, so dass die Gruppenmitglieder lernen sich einzubringen und sich gegebenenfalls zurückzustellen. Die Fähigkeiten werden besonders dann wichtig, wenn die gesamte Gruppe, wie es hier der Fall ist, sich neu in ein Thema einarbeiten muss. Die Themenbereiche bezogen sich nicht auf rein informatische Probleme, sondern umfassten viele Fachbereiche, so dass die SchülerInnen sich intuitiv mit anderen Fragestellungen auseinandersetzten.

Die Räumlichkeiten am Luisengymnasium sind für einen Informatikunterricht mit Testfläche für die Roboter nur begrenzt praktisch. Der Unterricht findet in den Physik- und Chemieräumen der Schule statt, so dass die Tische feststehend in mehreren Reihen aufgestellt sind. Die Lehrkraft kann in diesem Fall schwer den Überblick über die Geschehnisse auf dem Notenbook und die Arbeitsschritte der einzelnen Gruppen überprüfen. So kann es vorkommen, dass eine Gruppe Spiele spielt, anstatt an dem jeweiligen Problem zu arbeiten. Zudem ist es schwer sich durch die Stuhlreihen zu den Arbeitsplätzen der SchülerInnen zu bewegen, um bei Fragestellungen der Gruppe zu helfen. Optimal wäre ein Klassenraum in dem die Tische in einem Kreis oder Halbkreis aufgestellt wären, so dass in der Mitte eine Testfläche für Roboter oder andere Präsentationen frei wäre und der Lehrer von allen Seiten Einblick auf die Notebooks und damit auf die Arbeitsweise der SchülerInnen hätte. Die Tische müssten dabei so platziert sein, daß die SchülerInnen mit dem Rücken zum Raum sitzen, damit der Blick des Betreuers frei auf die Arbeitsflächen der Gruppen ist. Dies ist nicht nur nützlich um den Arbeitsfortschritt der Gruppen im Blick zu behalten, sondern kann typischen Unfällen vorbeugen, zum Beispiel unbedachtes Abstellen der Arbeitsmaterialien und damit die Gefahr, dass Notebook oder Roboter oder Baumaterial beschädigt werden. Gerade wenn viele Gruppen miteinander diskutieren kann es oft zu unbeabsichtigten Unfällen kommen. Zur Zeit testen die SchülerInnen ihren Roboter auf dem Gang vor dem Klassenraum oder fahren um das Lehrerpult herum. Der Platz zum Austesten von Ideen ist sehr beschränkt, so dass es nicht möglich ist, dass mehrere Gruppen gleichzeitig ihr Programm austesten. Hiermit verbunden ist die Gefahr, dass die Roboter beschädigt werden, wenn mehrere Gruppen gleichzeitig Testläufe starten. Oftmal fahren viele Roboter durcheinander, begleitet von den Gruppen. Das Risiko in dieser Arbeitsweise besteht darin, dass Gruppenmitglieder versehentlich auf einen Roboter treten könnten, wenn sie rückwärts gehen oder einem anderen Roboter ausweichen. Es wäre nötig einen geeigneten Raum für solche Experimente zu finden. Daran schließt sich ein anderes Problem an. In den Chemie -und Physikräumen befinden sich Stahlschränke in denen Notebooks und Roboter gelagert werden, um sie sicher aufzubewahren. In einem anderen Raum sind solche Sicherheitsvorkehrungen nicht vorhanden, d.h. die SchülerInnen müssten ihren Roboter jedesmal von einem Raum zum anderen bringen und es würde wertvolle Unterrichtszeit verloren sein.

Es gibt die Möglichkeit einer mobilen, zerlegbaren Teststrecke für die Roboter, die von der HAW an das Luisengymnasium ausgeliehen wurde. Leider konnte diese Teststrecke bislang nicht eingesetzt werden, da der Platz im Klassenzimmer zu begrenzt ist. Einige Aufgaben-

stellungen, zum Beispiel das Fahren auf einer schwarzen Linie, grössere Touren fahren oder ähnliches ist zum Teil aus diesem Grund zur Zeit nicht möglich.

Ein weiteres Problem besteht darin, dass die Microsoft Robotics Studio nicht ausgereift ist und sich verändert. Die Studenten und SchülerInnen müssen sich den neuen Begebenheiten anpassen, so kann es zum Teil zu Verzögerungen kommen, die den Unterricht beeinflussen. Dies ist ein typisches Informatikproblem und vermittelt so den SchülerInnen einen weiteren Einblick in die Informatik und dessen Herausforderungen. Viele SchülerInnen fassen es als solche Herausforderung auf und warten gedultig, bis eine Lösung gefunden wurde. Diese Zusammenarbeit zwischen Lehrkraft, Studierenden und SchülerInnen bringt allen Seiten viele Vorteile. Sie lernen nicht nur den Umgang mit neuen Systemen und deren Problemen, sondern lernen mit neuen Herausforderungen umzugehen. Eine Erfahrung, die in einem typischen Schulalltag, in dem meist Aufgaben und Lösungen von der Lehrkraft vorgegeben werden, ist dies eine wesentliche Erfahrung. Eigene und andere Stärken und Schwächen zu nutzen, um gemeinsam ein Projekt erfolgreich zu bestehen ist kein alltägliches Ergebnis eines Unterrichtes. Solche Erlebnisse bereiten die SchülerInnen unbewusst auf viele Ansprüche ausserhalb und innerhalb der Schulstrukturen vor. Ein weiterer Vorteil dieser Problematik ist die Ausreifung von des Microsoft Robotics Studio im Unterricht. Es wird direkt im Unterricht ( und dessen Anforderungen an ein Programm ) weiterentwickelt. Zudem nehmen die Schüler/-Innen aktiv an dieser Entwicklung teil, dies bringt Vorteile für alle beteiligten Seiten indem ein Perspektivenwechsel stattfindet. Die Aufmerksamkeit kann auf Schwierigkeiten gelenkt werden, die sonst möglicherweise nicht beachtet werden würden.

Das Projekt am Luisengymnasium ist trotz einiger Schwierigkeiten erfolgreich. Viele Lernbegriffe und Lerninhalte wurden unbemerkt vermittelt und die Aufmerksamkeit von SchülerInnen auf den Bereich Informatik gelenkt. Die Klasse ist an neuen Aufgaben interessiert und entwickelt von sich aus neue Fragen und Aufgabenstellungen, an denen sie weiterarbeiten.

## 7 Der Unterricht am Kurt-Körper-Gymnasium

Das Kurt-Körper-Gymnasium ist im Bereich Informatik bereits aktiv, d.h. es ist Informatikunterricht schon in der Sekundarstufe I vorhanden und es konnten schon einige Erfahrungen durch das Projekt e-Truck in der Oberstufe gesammelt werden. Bislang wurde in der Sekundarstufe I noch nicht mit Robotern gearbeitet, es wurde erst in der Oberstufe in diesem Themenbereich angesetzt.

Die Klasse besteht zu gleichen Teilen aus Jungen und Mädchen. Sie besitzen keine Vorerfahrungen im Programmieren, so dass sie an dieses Thema herangeführt werden müssen.

Die ersten Unterrichtseinheiten beschäftigten sich mit der Einführung in das Thema Roboter. Es wurden verschiedene Roboter vorgestellt und ihre Funktion und ihre Gestalt mit den SchülerInnen zusammen geklärt. Auffällig war, dass die Klasse den Roboter vermenschlicht, er ist je nach Lage zum Beispiel traurig oder fröhlich, böse oder gut. Die Programmierung von Robotern wurde durch praktische Übungen erklärt: eine Person stellt sich in den Raum und wird mit einfachen Befehlen durch diesen gelenkt. Zunächst wird nur der Befehl „gehe\_vor“ verwendet, wobei ein „gehe\_vor“ nur einen Schritt der Person darstellt. Die Klasse befiehlt dies so lange, bis der menschliche Roboter gegen die Wand läuft. Nun wird der Befehl „stop“ eingesetzt. Nach und nach folgen durch verschiedene Aufgabenstellungen die Befehle „nimm\_auf“, „gib\_ab“ und „drehe\_links“. Alle Befehle wurden an die Tafel geschrieben, so dass sie ständig präsent waren und später von der Klasse abgeschrieben werden konnten. Die Schüler besitzen ein Heft in dem sie alle Arbeitsschritte und programmierteile festhalten, damit sie ein Nachschlagewerk erhalten und sich auf ihre Klausur vorbereiten können. Die verwendeten Befehle waren eine Hinführung auf das Programm „WinNiki“, eine Robotersimulation, in der die SchülerInnen erste Versuche in der Programmierung starteten. Nachdem einige Erfahrungen mit dem Programm „WinNiki“ gesammelt wurden, wurden die Lego Mindstorm NXT zusammengebaut. Dabei wurde die Bauanleitung von Lego aus dem Internet benutzt, so dass alle Roboter das gleiche Bauschema aufweisen. Die Gruppenstärke der SchülerInnen besteht aus 2-3 Personen, die gemeinsam mit einem Roboter an einem Computer arbeiten. Das Kurt-Körper-Gymnasium besitzt zwei Computerräume, so dass in diesem Fall keine Notebooks angeschafft wurden. Nach der Fertigstellung der Roboter wur-



den mit dem Testprogramm von Lego erste Funktionen des Roboters getestet, indem sie eine bestimmte Zeiteinheit lang vor und zurück fahren.

Die folgende Aufgabe bestand darin den Roboter ein Quadrat fahren zu lassen. Einige der SchülerInnengruppen haben diese Aufgabe weitgehendst gelöst, wobei die meisten Roboter einen Kreis statt quadratisch fahren. Ein Problem an dieser Aufgabe ist der rechte Winkel, den der Roboter ausfahren muss. Das Arbeiten an dieser Aufgabe macht den SchülerInnen sehr viel Spass, sie probieren viele Lösungsvarianten aus und erproben die Möglichkeiten mit dem Roboter. Den Unterricht begleiten neben der Lehrerin, eine Studentin und eine Schülerin aus der Oberstufe. Bei Fragestellungen geben die Begleiter Anregungen, aber keine Lösungen weiter, die Arbeitsgruppen müssen sich möglichst selbst ihre Lösungen erarbeiten, damit sie im Umgang mit informatischen Problemen sicher werden und sie später auf andere Bereiche übertragen können.



Abbildung 7.1: SchülerInnen des Kurt-Körper-Gymnasiums

Als weitere Aufgabenstellung folgte das Fahren über eine Brücke, die auf einer Übungsplattform befestigt wurde. Bei dieser Aufgabe traten verschiedene Probleme auf, unter anderem konnte die Brücke nicht überquert werden, weil ein Sensor an diese stiess. Die Gruppen haben unterschiedliche Arbeitslevel, so dass die Aufgaben individuell gestaltet werden, damit fortgeschrittene Gruppen genügend gefördert werden. Bei Problemen helfen sich die SchülerInnen gegenseitig, bevor sie Anregungen durch die Lernbegleiter suchen.

Zur Zeit bereiten sich die SchülerInnen auf die Nacht des Wissens vor, um etwas vorführen zu können. Dieses Ereignis motiviert die Gruppen zusätzlich und sie arbeiten eng zusammen, um ihre Roboter vorstellen zu können.

Das Kurt-Körper-Gymnasium programmiert mit NXC und nicht wie das Luisengymnasium in C#. NXC ist das Nachfolgemodell von NQC, das sich an die erste Generation der Lego

Roboter angelehnt hat. Die Computer des Kurt-Körper-Gymnasiums besitzen nicht die Ausstattung, die für die Simulation, die am Luisengymnasium verwendet wurde, nötig wäre. Aus diesem Grund wurde hier zur Einführung das Programm „Winikki“ benutzt. Es war sinnvoll mit diesem Programm anzufangen, da sich NXC und die Befehle für „Winikki“ ähneln. Es liegt also eine gleiche Arbeitsweise wie am Luisengymnasium vor, nur mit unterschiedlichen Programmieransätzen. Der Vorteil von NXC besteht zudem darin, dass viele anfängliche Probleme schon geklärt wurden, die in der Arbeit mit C# noch bestehen. C# hat den Vorteil, dass es eine Sprache ist, die sehr ausbaufähig ist und so noch in folgenden Jahrgängen benutzt werden kann, während NXC auf den ersten Blick einfacher anmutet und so der Lehrkraft erste Hemmungen im Umgang mit ihr nimmt.

## 8 Projektablauf

Der geplante Ablauf des Projektes bestand zunächst darin, dass am Luisengymnasium ein Konzept entwickelt wird, das später auf das Kurt-Körber-Gymnasium übertragen werden kann. Tatsächlich wurde bereits im März 2007 mit dem Projekt am Kurt-Körber-Gymnasium begonnen. Diese Veränderung der geplanten Struktur des Projektes hat den Vorteil, dass an zwei Schulen unterschiedliche Ansätze getestet werden können. Jeder Ansatz beinhaltet unterschiedliche pädagogische und informatische Inhalte, deren Gegenüberstellung interessante Resultate mit sich bringt. Am Luisengymnasium wurde der Unterricht mit einer Simulation begonnen, die sehr komplex war und der Struktur von Computerspielen ähnelten. Das Kurt-Körber-Gymnasium dagegen begann mit einer rollenspielerisch Einführung, in der die SchülerInnen die Funktionen eines Roboters übernahmen und dann zu einer einfach strukturieren Simulation übergingen. Beide Einstiege in das Thema Roboter hatten ihren Reiz für die Schüler, während am Kurt-Körber-Gymnasium die schauspielerische Variante der Roboter für Spass sorgte, faszinierte die Computerspielerische die SchülerInnen am Luisengymnasium. In beiden Formen des Unterrichtes konnte die Klasse Elemente aus ihrer Freizeit und alltäglichen Umgebung wiederfinden und damit eine besser Bindung zum Unterrichtsgeschehen aufbauen. Der Spass an der Arbeit mit den Robotern, sei es in der Simulation oder mit den Mindstorm NXT, wurde durch die Pflicht der Benotung nicht beeinträchtigt.

Unterschiede ergaben sich nicht nur in der Einführung in das Thema, sondern auch in der Programmiersprache. Die Sprache C# ist eine objektorientierte Sprache, die zum einen sehr weit verbreitet ist und zum anderen durch ihre Ähnlichkeit zu anderen objektorientierten Sprachen in höheren Jahrgangsstufen den Umstieg auf andere Sprachen erleichtert. Die Lehrer am Luisengymnasium besaßen einige Vorkenntnisse in Java, so dass diese Kenntnisse optimal genutzt werden konnten. Am Kurt-Körber-Gymnasium wurde NXC angewendet. NXC ist die Nachfolgesprache von NQC, die mit dem Vorläufer des Lego Mindstorm NXT benutzt wurde. Dementsprechend waren viele Schwierigkeiten schon vorab beseitigt und die Klasse konnte einwandfrei mit dem Programm arbeiten. C# ist eine neue Programmiersprache für den Lego NXT. Das bedeutet, dass es einige Schwierigkeiten mit der Anwendung gab, die zum Teil spontane Wendungen im Unterricht nach sich zogen. Zum Teil mussten Inhalte improvisiert und geplante Unterrichtseinheiten umgestellt werden.

Beide Gymnasien verfolgten unterschiedliche Konzepte der Wissensvermittlung. Am Luisen-

gymnasium wurden die SchülerInnen nach einer kurzen Einführung Programmierumgebung in die Aufgabenstellung entlassen. Nach und nach wurden weitere Elemente der Informatik erschlossen und nach Bedarf der SchülerInnen weiter vertieft. Die Lehrkraft zog sich weitgehendst aus dem Geschehen und gab selten Erklärungen ohne direkten SchülerInnen Bedarf vor. Am Kurt-Körber-Gymnasium wurden die SchülerInnen direkt auf das Thema hingeführt und dann in der Aufgabenbearbeitung begleitet. Jede Stunde gab zunächst einen weiteren Arbeitsschritt vor, der zusammen mit der Klasse erarbeitet wurde. In der zweiten Phase des Unterrichtes konnten die SchülerInnen dann entweder an dieser neuen Aufgabe arbeiten, vorheriges wiederholen oder eigene Aufgaben verfolgen. Die Klasse hielt die einzelnen Arbeitsschritte schriftlich fest, da später eine Klausur darüber geschrieben wurde. Die Klasse des Luisengymnasiums verfasste Protokolle der einzelnen Unterrichtseinheiten die später benotet wurden. Dabei machten sie sich während des Unterrichtes selten Stichpunkte, so dass diese Protokolle rein aus dem Gedächtnis entstanden und damit ihren Lerneffekt gleich darstellten.

Diese verschiedenen Vorgehensweisen der Schulen waren sinnvoll, um einen direkten Vergleich der unterschiedlichen Ansätze zu ermöglichen. Aus diesem Grund musste von der ursprünglichen Planung abgewichen werden.

## **Erfahrungsbericht aus dem Luisengymnasiums, Werner Baum**

Die Praxisphase der Unterrichtseinheit gliederte sich in die folgenden Abschnitte:

- Arbeiten mit dem Microsoft-Robotics-Studio und der Programmiersprache C# in der Simulationsumgebung
- Aufbau der Lego-NXT-Roboter
- Steuerung der Roboter über das Microsoft-Robotics-Studio

Die Praxisphase wurde eingeleitet mit einem Vortrag von Prof. Dr. Kai von Luck der einen Bogen spannte von der Verwendung von Mikrochips in den Gegenständen des täglichen Lebens bis hin zum Einsatz von Robotern. Den Umgang mit der Simulationsumgebung erlernten alle Schüler sehr schnell. Die studentischen Mitarbeiter stellten dazu die nötigen Handgriffe und Befehlsfolgen kurz vor, die Schüler vollzogen diese dann nach. Bei der Bearbeitung der gestellten Programmieraufgaben zeigte sich aber sehr schnell ein differenzierteres Bild. Während eine Gruppe von zwei Mädchen in kürzester Zeit mit einem überraschend knappen Programm die Aufgabe meisterte, den virtuellen Roboter ohne „Crash“ um ein Hindernis fahren zu lassen, ergaben sich bei anderen Gruppen unterschiedlichste Programmier- und Verständnisprobleme. Zum Teil gelang es

den Schülern durch Hilfestellung untereinander die Probleme zu lösen, in anderen Fällen war der Rat der Lehrkraft und die Hilfestellung der Mitarbeiter der HAW nötig. Schwierig bei dieser äußerst eigenständigen Gruppenarbeit ist, dass beim gleichzeitigen Auftreten von Schwierigkeiten in mehreren Arbeitsgruppen diese in eine Warteschleife gelangen, die den Fortschritt der Arbeit in dieser Gruppe behindert, Frustration erzeugen kann und auch ein Auseinanderlaufen der Wissensstände des Kurses zur Folge haben kann. Bisher ist es gelungen, eine zu starke Differenzierung innerhalb der Schülergruppe zu verhindern. Die Motivation der Schüler durch die Simulationsumgebung ist allerdings auch außerordentlich hoch, nicht zuletzt durch ihre Ähnlichkeit mit einem Computerspiel. Dadurch war die Gruppe bisher durchaus bereit, sich sehr weit auf die Auseinandersetzung mit dem theoretischen Hintergrund einzulassen und auch hartnäckig bei auftretenden Schwierigkeiten am Programm weiterzuarbeiten. Einzuräumen ist, dass sich der zunächst aufgestellte Katalog des theoretischen Hintergrundwissens als zu anspruchsvoll erwies. Der Begriff der Variablen als Speicher für Werte wurde noch akzeptiert, der Sinn der verschiedenen Typen von Variablen blieb für die Schüler jedoch zunächst zum großen Teil im Dunklen. Brav wurden auch die verlangten Nassi-Shneiderman-Diagramme angefertigt, die Nützlichkeit dieses Werkzeuges zur Programmanalyse erschloss sich den Schülern aber bisher nicht vollständig. Auch der Begriff der Funktion war bisher nur schwer zu vermitteln. Dabei können die Schüler Variablen und Funktionen in ihren Programmen rein pragmatisch sinnvoll benutzen und als Werkzeug zur Problemlösung einsetzen. Wichtig wird in der Fortsetzung der Unterrichtseinheit sein, durch ein fortgesetztes Hin- und Herschalten zwischen praktischer Arbeit und theoretischer Reflektion auch das theoretische Verständnis weiterzuentwickeln, ohne den spielerischen Charakter zu verlieren. Gerade hier, das zeigt die bisherige Erfahrung, liegen die Stärken des Projektes.

Beim praktischen Aufbau der Roboter mit den Lego-Bausätzen zeigte sich der hohe Aufforderungscharakter des Materials. Die Schüler und Schülerinnen waren mit Feuereifer bei der Sache und konstruierten ausgesprochen kreativ; sie lösten sich dabei sehr schnell völlig von den von Lego mitgelieferten Vorlagen und entwickelten eigene, unterschiedliche Modellvarianten, die den Vorgaben: ein möglichst niedriger Schwerpunkt und die Fähigkeit auf eine Rampe zu fahren, Rechnung tragen sollten. Nicht alle Konstruktionen erwiesen sich als uneingeschränkt brauchbar; die Vorzüge und Schwachstellen der entwickelten Roboter diskutierte Prof. v. Luck aus der Ingenieurssicht in einer Doppelstunde mit den Schülern. Interessant ist in diesem Zusammenhang die Bemerkung einer Schülerin: „Lego-Technik kannte ich bisher nur vom Einwickeln der Schachteln als Geschenk, wenn ich zu einem Jungen zum Geburtstag eingeladen wurde.“ Die „gefühlte Kompetenz“ der Jungen war hier zunächst

durch den vertrauteren Umgang mit dem Technikbaukasten größer, ein Faktum, das in vielen Bereichen des naturwissenschaftlichen- und Technikunterrichtes festzustellen ist. Dennoch konstruierten auch die Mädchen ausdauernd und erfolgreich mit. Dieses nicht zuletzt deshalb, da sie durch ihre lauffähigen Simulationsprogramme in der Einführungsphase eben doch Ansätze eines positiven Selbstkonzeptes in diesem Bereich entwickeln konnten.

Die Roboter über die Bluetooth-Schnittstelle anzusprechen gelang im Wesentlichen problemlos. Schwierigkeiten ergaben sich aber beim Übertragen der Signale der Sensoren vom Roboter via Bluetooth auf den Rechner; z.T. wurden hier Signale völlig verschluckt oder aber zeitlich so stark verzögert, dass eine sinnvolle Reaktion nicht programmierbar war.

Die Ursache dieses Problems muss jetzt gefunden und behoben werden. Für die Schüler waren diese letzten Schulstunden, in denen sich dieses unberechenbare Verhalten der Roboter zeigte, sehr unbefriedigend. Deutlich zeigt sich an dieser Stelle der Forschungscharakter des gesamten Vorhabens. In dieser Form ist das Projekt noch nicht praxistauglich.

Es wäre außerordentlich schade, wenn sich bei der Analyse der Signalübertragung zeigen sollte, dass hier ein grundsätzliches Manko der verwendeten Hardware liegt, hat sich doch der Einstieg über die Simulationsumgebung und die Programmierung mit dem Robotics-Studio unter der Verwendung von C# als sehr tragfähig und uneingeschränkt positiv erwiesen.<sup>1</sup>

## **Erfahrungsbericht aus dem Kurt-Körber-Gymnasium, Christine Klemm**

Am Kurt-Körber-Gymnasium ist, abweichend von der Stundentafel für Gymnasien in Hamburg, das Fach Informatik ab der 8. Klasse Pflicht für alle Schüler. Für die Teilnahme an diesem Projekt wurde eine 8. Klasse ausgewählt, so dass die Schülerinnen und Schüler noch keine Vorkenntnisse mitbrachten. Im ersten Halbjahr erfolgte eine Einführung in die Grundlagen der Textverarbeitung und in das Arbeiten mit Präsentationssoftware. Zu Beginn des zweiten Halbjahres wurde zunächst die Herkunft des Begriffs "Roboter" ausführlich geklärt und anhand von Alltags-Algorithmen an den Algorithmusbegriff herangeführt. Danach erfolgte eine sechsstündige Einführung in das Programmieren anhand eines speziell für den Anfangsunterricht geeigneten Robotermodells (WinNiki).

---

<sup>1</sup>Werner Baum

Es ermöglicht das Kennenlernen von Befehlsabläufen, die Abfrage einfacher Sensoren und das Formulieren von Wiederholungsanweisungen (ohne Zählschleife) und Fallunterscheidungen.

Für das Arbeiten mit den NXT-Robotern kamen verschiedene Oberflächen infrage. Das Robotics Visual Studio von Microsoft konnte nicht verwendet werden, da die vorhandenen Grafikkarten in den Schulcomputern nicht die gewünschten Anforderungen erfüllten. Erst im Herbst dieses Jahres wird die Anschaffung neuer Notebooks möglich werden. Aus diesem Grund fiel die Entscheidung zugunsten des Einsatzes von NXC, der Nachfolgeumgebung von NQC. Auch diese lief zunächst nicht problemlos, da sie die Installation eines Treibers für die Roboter benötigt, der in der LEGO-Software enthalten ist. Diese konnte nur unter Schwierigkeiten installiert werden, da nur XP Service pack 1 auf den Schulrechnern vorhanden war und die Software eine Schreibberechtigung in den Ordner „Eigene Dateien“ verlangt. Dieser war jedoch aus pädagogischen Gründen vor den Schülern verborgen worden, damit sie die Rechner nicht verbotenerweise mit Spielen oder anderem belasten.

Für die Schülerarbeit stehen 10 Baukästen für 27 Schülerinnen und Schüler zur Verfügung. Sie durften ihre Gruppenzugehörigkeit frei wählen, wobei keine gemischt-geschlechtliche Gruppe gebildet wurde.

Zur Zeit arbeiten eine beamtete Lehrkraft (Frau OStR' Klemm), eine Pädagogin, die das Projekt begleitet, und eine Schülerin aus der 12. Klasse am Projekt mit. Die Schülerin hat donnerstags in den ersten beiden Stunden keinen Unterricht und ist freiwillig und unentgeltlich anwesend, betreut die Achtklässler bei technischen Schwierigkeiten und überwacht die Aus- und Rückgabe der Robotersets. Zeitweilig ist auch eine Informatikerin mit eingebunden, die als Quereinsteigerin in den Schuldienst im Herbst mit dem Vorbereitungsdienst beginnen möchte. Wir hoffen, dass sie unserer Schule zugewiesen wird und dann im Rahmen des selbständig zu erteilenden Unterrichts in das NXT-Projekt einsteigen wird.

Die Schülerinnen und Schülern waren zunächst alle sehr begeistert. Sie bauen sehr konzentriert und fleißig. Zwei Gruppen waren frustriert von dem Gefühl, dass Teile im Bausatz fehlen, obwohl auch diese Schüler original verpackte Sätze erhalten hatten.

Mit der Zeit kristallisierten sich einige Schwierigkeiten heraus: Die NXT-Bausätze enthalten sehr viele Kleinteile und sind deutlich unübersichtlicher als die alten LEGO-Bausätze (die Roboter sind aber wesentlich stabiler, das ist

ein Vorteil). Die Schüler suchen sehr lange nach dem richtigen Teil, das als nächstes eingebaut werden muss; die Doppelstunden wirken sehr kurz.

Die Gesamtgruppe wird zum Basteln auf zwei benachbarte Räume verteilt. Eine angemessene Betreuung wäre durch eine einzige Lehrkraft kaum möglich. Ein Vorteil ist, dass den Schülern zwei große Arbeitsflächen (in der jeweiligen Raummitte) zur Verfügung stehen.

Die Durchführung einer Doppelstunde beginnt immer mit einem Plenum, in dem der Stand der einzelnen Gruppen bekannt gemacht wird. Danach werden Informationen über das Stundenziel an die Schüler gegeben und es erfolgt eine gemeinsame Erarbeitung der Sachinformationen, die benötigt werden, um das Stundenziel zu erreichen.

Dann erfolgt die Ausgabe der LEGO-Kästen und die Verteilung auf die beiden Räume. Da die Gruppen sehr unterschiedliche Arbeitsgeschwindigkeiten haben, ist schon ein deutliches inhaltliches Auseinanderdriften zu beobachten.

Etwas ungünstig hat sich auf unseren Zeitplan ausgewirkt, dass wegen eines Frankreichaustausches am Anfang des Halbjahres einige Schüler zweimal gefehlt haben und zu einem Termin französische Gast Schüler zusätzlich untergebracht werden mussten.

Die Gruppe bekam am Anfang als ferne Zielsetzung, eine Vorführung in der HAW am 9. Juni im Rahmen der "Langen Nacht der Wissenschaften zu gestalten", was einige Schüler zusätzlich stark motiviert.

Aufgrund der vielen positiven Erfahrungen soll das Projekt im nächsten Schuljahr unbedingt weitergeführt werden, wobei über die Klassenstufen erst nach der Lehrerzuweisung entschieden werden kann.<sup>2</sup>

---

<sup>2</sup>Christine Klemm



## 9 Bewertung des Ansatzes

Die SchülerInnen haben in den bisherigen Unterrichtseinheiten einen guten Einblick in die Programmierung und die damit zusammenhängenden Anforderungen erhalten. Für die meisten SchülerInnen war es die erste Erfahrung im Umgang mit Programmierung und Robotern, so dass viele Grundbegriffe erarbeitet werden mussten. Der Wissensstand der SchülerInnen driftete während des Unterrichtes auseinander, so dass der Lernbegleiter sich ständig neu auf Situationen und Probleme einstellen musste. Für den Lehrer/-Innen eine Herausforderung, da zum einen ein Rollenwechsel zum Lernbegleiter stattfand, zum anderen, weil es zum Teil ein unbekanntes Terrain war auf dem gearbeitet wurde. Besondere Situationen entstanden am Luisengymnasium, da hier C# verwendet wurde und noch einige anfängliche Probleme ausgemerzt werden mussten. So konnte es vorkommen, daß spontan neue Elemente eingeführt wurden, mit denen selbst die Lehrkraft noch nicht vertraut war. Unterstützung wurde durch die Studenten geleistet, die jede Stunde anwesend waren, um tatkräftig den Unterricht zu begleiten. Während des Unterrichts wurde den SchülerInnen viel Freiraum für eigene Erfahrungen und den spielerischen Umgang mit der Simulation gewährt. Der Unterricht sollte an viele Elemente des selbstgesteuerten Lernens beinhalten, damit die SchülerInnen möglichst eigenständig arbeiten. Für die Lehrkraft ist der oben erwähnte Wechsel von traditioneller Lehrtätigkeit zum Lernbegleiter schwierig, da die Kontrolle des Lernverhaltens nicht mehr direkt von ihm gesteuert werden kann. Beim selbstgesteuerten Lernen ist nicht nur die Freiwilligkeit wichtig, sondern auch die Verbindlichkeit. Jeder Einzelne der Gruppe kennt seine Bedeutung innerhalb der Aufgabenstruktur und ist sich darüber im Klaren, dass das Ergebnis des jeweiligen Lernprozesses eingefordert werden kann. Diese Verbindlichkeiten treten dem Gefühl, dass es egal sein könnte, ob jemand mitarbeitet oder nicht, entgegen, gleichzeitig weckt und steigert es die „Könnenserfahrung“ (Gudjons) der SchülerInnen und damit ein wichtiges Element für die Selbstwirksamkeit des Lernens. Diese Kombination von Lust am Lernen, das eigene Können erweitern und die Erfolgserlebnisse, die aus dieser Arbeit in Selbstständigkeit entstehen, lassen das Selbstvertrauen und so den Spass an neuen Herausforderungen wachsen. Diesen Spass am Lernen konnte in jeder Klasse nachgewiesen werden. Viel SchülerInnen des Kurt-Körber-Gymnasiums verbrachten ihre Pausen und Mittagszeit freiwillig damit weiter an den Robotern zu bauen und am Programm zu arbeiten. Die Absprachen innerhalb der einzelnen Gruppen funktionierten gut, sie ergänzten sich sowohl beim Zusammenbauen der Roboter als auch beim Programmieren und Testen. Bei Verständnisproblemen orientierten sie sich zunächst im Klassenverband, bevor sie sich Un-

terstützung durch die Lehrkraft holten. Ein abschließender Höhepunkt des Projektes war die Nacht des Wissens, in der die SchülerInnen ihre Roboter und selbst erarbeitete Aufgabenstellungen der breiten Öffentlichkeit präsentierten. Die Vorbereitungen auf diese Nacht motivierten die Klassen sowie Lehrkräfte zusätzlich, da es unter den SchülerInnen wie ein Wettbewerb gehandelt wurde. Die Nacht des Wissens war ein grosser Erfolg für alle Beteiligten, da die Begeisterung der SchülerInnen auf die Besucher der Veranstaltung überging. Die SchülerInnen präsentierten unermüdlich ihre Aufgabenstellungen, ihren Roboter und gaben bereitwillig Auskunft über ihr Projekt.



Abbildung 9.1: Präsentation des Kurt-Körber-Gymnasiums

Informatikunterricht stellt einen Grundbaustein in der Medienkompetenz dar, weil in diesem Fachbereich die Wechselwirkung und die Struktur von Informatiksystemen analysiert wird und die Teilnehmer eigene mediale Produkte und Systeme gestalten sollen. In diesem Punkt ist ein fächerübergreifender Unterricht sehr wichtig, da hier die Werkzeuge zur Strukturierung und Verknüpfung von verschiedenen Informationen bereitgestellt werden. Die anwendungsorientierte Nutzung vom erworbenen Wissen fördert die Wahrnehmung, Durchdringung und Bearbeitung von komplexen Problemstellungen. Informatik ist eine Wissenschaft, die sich stetig und schnell entwickelt, demzufolge treffen die Schülerinnen und Schüler in ihrem Unterricht auf sich wiederholende Konzeptionen und Wirkungsprinzipien, die ihnen ermöglichen sich rasch in die Nutzung neuer Systeme einzuarbeiten. Neben den handlungsorientierten Vorteilen des Faches Informatik bietet die Arbeit mit Robotern kreative Gestaltungsmöglichkeiten. Selbstständige Analyse, die Entwicklung und Beschreibung eigener Lösungsansätze, deren Reflexion und Verbesserungen fördern kreatives und schöpferisches Denken. Durch die komplexe Problemstellungen der Aufgaben und deren Lösung innerhalb eines bestimmten Zeitraumes ist ein arbeitsteiliges Vorgehen nötig. Sie lernen so das Kommunikation und Kooperationsfähigkeit wesentlich sind, um ein Projekt oder eine Aufgabe zu gestalten. Die Durchführung und Lösung einer Aufgabe durch Gruppenarbeit führt sie an Projektarbeit und die damit verbundenen Denkansätze hinsichtlich Planung, Entscheidungen treffen, Präzi-

sion, Aufteilung von Aufgaben und Einhaltung von Absprachen heran. Der Unterricht hilft den TeilnehmerInnen sich fachlich zu orientieren und vermittelt ihnen einen Vorsprung im späteren Bildungsverlauf. Sie erhalten wesentliche Grundlagen in Kommunikations- und Informationstechniken und Erfahrungen in typischen Problemlösungsstrategien, die ihnen die notwendige Basis für diverse Berufsfelder geben. Die Arbeit mit Robotern fördert bildhaftes Lernen, sowie es die Angst davor nimmt Neues auszuprobieren. Bei späteren, ähnlichen Aufgabenstellungen kann durch Schlüsselemente auf dieses Wissen zurückgegriffen werden. Diese Form des Lernens vertieft die gewonnen Erkenntnisse und bringt gleichzeitig eine Reflexion über bereits Erlerntes mit sich. Die Arbeit mit dem Roboter stellt eine Form des objektorientierten Lernens dar, da Lerninhalte sowie Lernergebnisse an ihn gebunden und gleichzeitig leicht auf andere Aufgabenstellungen übertragbar sind.

Die bisherigen Erfahrungen mit Informatikunterricht an Schulen bergen einige Hürden, die mit neuen Ansätzen, wie dem selbstgesteuertem Lernen und projektartigem Unterricht verbessert werden könnten. Bislang wurde im Unterricht meist Produktwissen erarbeitet und vertieft. Die Arbeit mit Robotern im Informatikunterricht vermittelt die Grundlagen der Informatik spielerisch, so dass die SchülerInnen ein umfassendes Wissen erwerben auf das sie später aufbauen können.



Abbildung 9.2: SchülerInnen präsentieren auf der Nacht des Wissens

## 10 Ausblick

Die Ansprüche des Projektes haben sich bislang in vieler Weise bestätigt und damit als erfolgreich erwiesen. Den SchülerInnen konnte in unterschiedlichen Ansätzen der teilnehmenden Schulen vielseitige Kompetenzen vermittelt werden. So haben sie einen Einblick in die Strukturen und Probleme der Programmierung und der Informatik erhalten können und eine neue Denkweise kennengelernt. Nach der Einführung der benötigten Begriffe und Kennenlernen der Programmierumgebung lernten die Schülergruppen selbstständig Aufgabenstellungen zu bearbeiten. Zur Lösung der Aufgaben, die im Unterricht gestellt wurden, mussten sie lernen systematisch zu denken und ihren eigenen Ansatz zur Problemlösung zu finden. Dies setzt einen längeren Prozess voraus, der von der Problemerkennung über die Bildung eines Modells hin zur programmiertechnischen Lösung führte. Hierbei ist ein hohes Maß an Abstraktionsfähigkeit nötig, eine Denkweise, die den SchülerInnen der Sekundarstufe I bislang nicht so vertraut ist.

Die SchülerInnen haben sich den einzelnen Aufgaben in Kleingruppen gestellt. Diese Gruppenarbeit ist in der Informatik als methodisches Vorgehen der Paarprogrammierung (engl. pairprogramming <sup>1</sup>) bekannt, eine Methode, die neben vielen anderen Vorteilen den kollaborativen Aspekt der Problemlösung besonders hervorhebt. Die SchülerInnen haben damit eine neue Arbeitsweise kennengelernt, die im späteren Bildungsweg sehr nützlich ist.

Durch die komplexe Problemstellungen der Aufgaben und deren Lösung innerhalb eines bestimmten Zeitraumes ist ein arbeitsteiliges Vorgehen nötig. Sie lernen so das Kommunikation und Kooperationsfähigkeit wesentlich sind, um ein Projekt oder eine Aufgabe zu gestalten. Die Durchführung und Lösung einer Aufgabe durch Gruppenarbeit führt sie an Projektarbeit und die damit verbundenen Denkansätze hinsichtlich Planung, Entscheidungen treffen, Präzision, Aufteilung von Aufgaben und Einhaltung von Absprachen heran.

SchülerInnen lernen besser wenn Aktivität und Eigenverantwortlichkeit in den Unterricht einfließt, da neben dem Wissenserwerb Selbstbewusstsein und Eigenständigkeit erworben werden. Diese Fähigkeiten sind ebenfalls im Rahmenplan für Informatik erwünscht und in die Ausführungen desselben eingebunden.

Als nächster Schritt ist für eine Verstetigung des curricularen Ansatzes dieses Projektes unter anderem die Einbindung von weiteren Mitgliedern des jeweiligen Schulkollegiums,

---

<sup>1</sup> Kent Beck

das die Informatik und Technik vertritt, wünschenswert. Dieses ist besonders in Schulen denkbar, die sich im Rahmen der aktuellen Debatten um eine Profilbildung ein mathematisch/technisches Profil geben wollen<sup>2</sup>.

In diesem Zusammenhang ist ein Versuch der Übertragbarkeit der bisherigen Erfahrungen aus diesen ersten Schulklassen auf andere SchülerInnengruppen wichtig. Der derzeitige Erkenntnisstand der Unterrichtsgestaltung lässt einige wenige Lücken offen, die in den letzten Phasen des Schulprojektes geschlossen werden sollen. Diese neuen Erkenntnisse werden in einer weiteren Klasse ausgebaut, um feststellen zu können, inwieweit das Projekt Lernvorteile gegenüber traditionellem Unterricht aufweist und ein frühzeitiges Ansetzen des Informatikunterrichts nützlich ist. Dies lässt sich nur dann erkennen, wenn das erweiterte Konzept unter gleichen Bedingungen in neuer SchülerInnenzusammensetzung durchgeführt wird. Die beiden bisherigen Ansätze der beiden Gymnasien werden beibehalten, um eine optimale Gegenüberstellung zu erreichen.

Diese Gegenüberstellung der verschiedenen curricularen Ansätze ist für die Ausbildung eines Schulprofils ein wesentlicher Schritt, da mit einer Profilbildung der Lehrer/-Innen Veränderungen unterliegt, die die Unterrichtsgestaltung einschließt. Die damit zusammenhängenden Unsicherheiten werden durch diese Unterrichtsbeispiele geschwächt und erleichtern weitere Arbeitsschritte in Richtung eines Profils.

Ein weiterer Schritt wäre verschiedene Lehrkräfte zusammenzubringen, um ihnen den Ansatz vorzustellen, ihren Umgang mit dem Unterrichtskonzept und ihren Eindruck der Wissensvermittlung und Wissenserwerb zu erfahren. Anhand dieser Informationen und der Rückmeldungen der SchülerInnen kann das Konzept weiterentwickelt werden.



Abbildung 10.1: Gäste der Nacht des Wissens

<sup>2</sup>Ein solches Profil findet sich z.B. in dem Konzept MINT-Schule (s.a. <[www.mint-ec.de](http://www.mint-ec.de)>).

**A Material**

# Roboter an Schulen - Materialiensammlung

Kai Rosseburg

3. Juli 2007, Hamburg

# Inhaltsverzeichnis

Inhaltsverzeichnis . . . . .	2
Abbildungsverzeichnis . . . . .	3
<b>1 Microsoft Robotics Studio</b>	<b>4</b>
1.1 Die Simulation . . . . .	4
1.1.1 Simulations-Dateien . . . . .	4
1.1.2 Gestaltung eigener Simulations-Welten . . . . .	5
1.2 Verhaltensbasierte Robotik . . . . .	8
1.2.1 Verhaltens-Dateien . . . . .	8
1.2.2 Roboter-Programmierung . . . . .	8



# Abbildungsverzeichnis

1.1	Die resultierende einfache simulierte Welt . . . . .	7
-----	--	---

# 1 Microsoft Robotics Studio

Das Microsoft Robotics Studio ist ein „Software Development Kit“ für die Entwicklung von Roboter-Programmen. Es stellt eine Sammlung von Diensten und Klassen bereit, mit denen man eine große Anzahl von unterschiedlichen Roboter-Plattformen programmieren kann.

Unterstützt werden bisher Roboter von Fischer Technik, Pioneer und LEGO.

Das Robotics Studio abstrahiert vollständig von der anzusprechenden Roboter-Hardware. So ist es ohne Probleme möglich ein Programm, das einen Pioneer ansteuert für einen LEGO NXT zu benutzen.

## 1.1 Die Simulation

Durch den integrierten Simulator ist es möglich auch ohne Roboter Programme zu entwickeln und zu testen. Der Simulator ist eine physikalisch korrekt berechnete 3D-Umgebung, die mit in DirectX programmiert und leicht anpassbar ist.

### 1.1.1 Simulations-Dateien

Die Quelltext- und Projekt-Dateien der simulierten Welt befinden sich im Ordner:

`C:\Microsoft Robotics Studio (1.0)\samples\SimulationTutorials\Tutorial2`

Die folgenden Tabelle gibt einen kurzen Überblick über die Dateien und ihre Bedeutung.

Dateiname	Bedeutung
AssemblyInfo.cs	Info-Datei der einzelnen Programmteile
Simulation Tutorial 2.htm	Ursprüngliche Beschreibung des Simulation Tutorial 2 von Microsoft
SimulationTutorial2.cs	Quelltext der simulierten Umgebung
SimulationTutorial2.csproj	Projekt-Datei
SimulationTutorial2.csproj.user	Benutzer-Datei
SimulationTutorial2.sln	Interne VisualStudio-Datei
SimulationTutorial2.sou	Interne VisualStudio-Datei

Für das Benutzen und Erstellen eigener Simulationen sind nur SimulationTutorial2.csproj und SimulationTutorial2.cs von Interesse. Nachdem die Projekt-Datei geöffnet wurde kann die die Quelltext-Datei SimulationTutorial2.cs bearbeitet werden.

### 1.1.2 Gestaltung eigener Simulations-Welten

Zur Gestaltung eigener Simulations-Welten stehen folgende Methoden zur Verfügung:

```
void AddSky()  
void AddGround()  
void AddCameras()  
void AddBox(Vector3 position, Vector3 dimensions, float mass, String name)  
void AddTexturedBox(Vector3 position, Vector3 dimensions, float mass, String name)  
void AddPioneer3DXRobot(Vector3 position)  
private void PopulateWorld()
```

*AddSky()* fügt der simulierten Welt einen Himmel hinzu.

*AddGround()* fügt der simulierten Welt einen einfachen, planen Boden hinzu.

*AddCameras()* fügt der simulierten Welt eine Kamera hinzu. Ohne Kamera kann die Simulation nicht betrachtet werden.

*AddBox(position, dimensions, mass, name)* ist die Methode mit der eine texturlose Box in die Simulation eingefügt werden kann. Die Methode erwartet vier Argumente *position* und *dimensions* vom Typ *Vector3*, *mass* vom Typ *float*, sowie *name* vom Typ *String*.

Mit *position* wird die Position in einem dreidimensionalen Koordinatensystem angegeben, mit *dimensions* wird die räumliche Ausdehnung bzw. Größe der Box festgelegt. Die Positions- und Größenangaben beziehen sich jeweils auf den Mittelpunkt der Box. Mit *mass* wird die Masse der Box festgelegt, je größer die Masse desto geringer ist die Wahrscheinlichkeit, dass der Roboter sie umfahren kann.

Jedes Objekt in der simulierten Welt braucht einen eindeutigen Namen, deshalb muss dieser mit *name* übergeben werden.

*AddTexturedBox(position, dimensions, mass, name)* erwartet die exakt gleichen Argumente wie *AddBox(position, dimensions, mass, name)*, der einzige Unterschied ist, dass mit *AddTexturedBox(position, dimensions, mass, name)* texturierte Boxen in die Welt eingefügt werden können. Momentan ist die Textur noch fest implementiert und kann nicht verändert werden.

*AddPioneer3DXRobot(position)* ist die Methode mit der ein Modell des Pioneer-Roboter in die simulierte Welt eingefügt werden kann. Die Methode erwartet das Argument *position* vom Typ *Vector3*, welches die Position des Roboters innerhalb der simulierten Welt festlegt. Das Modell des Pioneer verfügt über einen Kontaktsensor (Bumper) und einen LaserRangeFinder.

*PopulateWorld()* ist die Methode in der die simulierte Welt "zusammengebaut" wird.

Alle o.g. Methoden müssen innerhalb der *PopulateWorld()* aufgerufen werden. Eine beispielhafte *PopulateWorld()* könnte wie folgt aussehen:

```
private void PopulateWorld()
{
    AddSky();
    AddGround();
    AddCameras();

    AddBox(new Vector3(0.0f, 0.5f, 0.0f), new Vector3(5.0f, 1.0f, 0.2f), 1, "box");

    AddPioneer3DXRobot(new Vector3(0, 0.1f, -2));
}
```

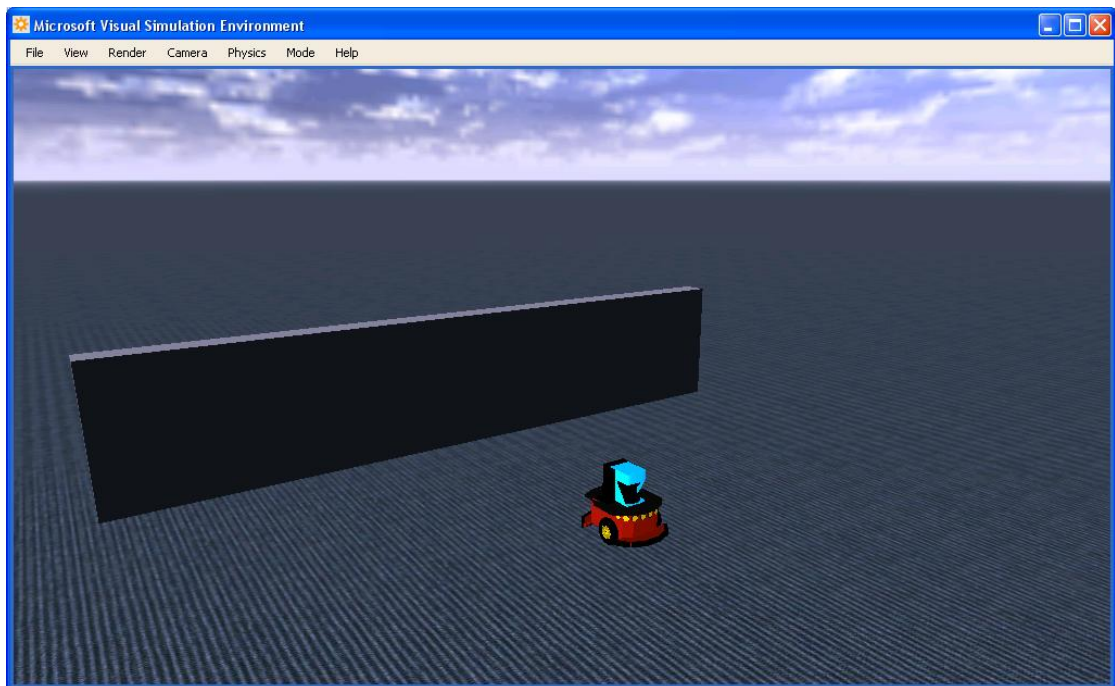


Abbildung 1.1: Die resultierende einfache simulierte Welt

## 1.2 Verhaltensbasierte Robotik

### 1.2.1 Verhaltens-Dateien

Die Quelltext- und Projekt-Dateien befinden sich im Ordner:

`C:\Microsoft Robotics Studio (1.0)\samples\RoboticsTutorials\Tutorial3\CSharp`

Die folgende Tabelle gibt einen kurzen Überblick über die Dateien und ihre Bedeutung.

Dateiname	Bedeutung
AssemblyInfo.cs	Info-Datei der einzelnen Programmteile
Robotics Tutorial 3.htm	Ursprüngliche Beschreibung des Robotics Tutorial 3 von Microsoft
RoboticsTutorial3.cs	Quelltext der simulierten Umgebung
RoboticsTutorial3.csproj	Projekt-Datei
RoboticsTutorial3.csproj.user	Benutzer-Datei
RoboticsTutorial3.sln	Interne VisualStudio-Datei
RoboticsTutorial3.sou	Interne VisualStudio-Datei

Um den Roboter steuern zu können sind nur RoboticsTutorial3.csproj und RoboticsTutorial3.cs von Interesse. Nachdem die Projekt-Datei geöffnet wurde kann die Quelltext-Datei RoboticsTutorial3.cs bearbeitet werden.

### 1.2.2 Roboter-Programmierung

Folgende Methoden können verwendet werden:

```
void RobotDrive(double left, double right)  
void MoveForward()  
void MoveBackward()  
void MoveLeft()  
void MoveRight()
```

```
int LaserLeft()  
int LaserFront()  
int LaserRight()  
bool FrontBumper()  
bool RearBumper()
```

*RobotDrive(left, right)* ist die Methode mit der der linke und der rechte Motor des Roboters angesprochen werden können. Die Methode erwartet zwei Argumente *left* und *right* vom Typ *double*, die die Geschwindigkeit des jeweiligen Motors darstellen.

Zu Beginn einer Unterrichtseinheit sind die Methoden *MoveForward()*, *MoveBackward()*, *MoveLeft()* und *MoveRight()* noch nicht implementiert. Die Implementierung kann von den Schülern geleistet werden, indem die o.g. Methode *RobotDrive(left, right)* benutzt wird.

Als sinnvoll haben sich folgende Werte für die einzelnen Methoden erwiesen:

<i>MoveForward()</i>	<i>RobotDrive(0.4, 0.4)</i>
<i>MoveBackward()</i>	<i>RobotDrive(-0.4, -0.4)</i>
<i>MoveLeft()</i>	<i>RobotDrive(0.0, 0.2)</i>
<i>MoveRight()</i>	<i>RobotDrive(0.2, 0.0)</i>

Diese Werte sind nur als Richtwerte zu verstehen und können im Laufe der Unterrichtseinheit den jeweiligen Bedürfnissen angepasst werden.

In der Simulation verfügt der Roboter über einen LaserRangeFinder, der zur Abstandsmessung verwendet wird. Der LRF deckt einen Bereich von 180° ab. Mit den Methoden *LaserLeft()*, *LaserFront()* und *LaserRight()* geben die Entfernung zum nächsten linken, mittigen oder rechten Hindernis in mm zurück.

Sowohl in der realen als auch in der simulierten Welt verfügt der Roboter über einen Kontaktsensor, den sog. Bumper (in der simulierten sind es sogar zwei - ei-

ner vorne und einer hinten). Die Methoden *FrontBumper()* für den vorderen und *RearBumper()* für den hinteren Bumper geben einen Wahrheitswert zurück.

"Bumper ist gedrückt."  $\implies$  WAHR

"Bumper ist nicht gedrückt."  $\implies$  FALSCH



# Literaturverzeichnis

- [fraunhofer:2005 2005] (Hrsg.): *Fraunhofer Institut Autonome Intelligente Systeme*. 2005.  
– URL <http://www.ais.fraunhofer.de>
- [fraunhofer:2006 2006] (Hrsg.): *Regelwerk RobCup Junior Rescue League*. 2006.  
– URL <http://alex.fraunhofer.de/zeno/web/Regelwerk+RoboCup+Junior+Rescue+League+2006-01.pdf?action=openattachment&id=16657&attachmentid=6597&rootid=15465>
- [Beck 2000] BECK, Kent: *Extreme Programming*. München, Germany : Addison-Wesley, 2000
- [Gudjons ] GUDJONS, Herbert: *Selbstgesteuertes Lernen der Schüler:Fahren ohne Führerschein? in Pädagogik,55.Jahrgang, Heft 5, Mai 2003*
- [Guerin 2006] GUERIN, Philippe: *Raumanalyse*. HAW Hamburg, 2006
- [Gunnerson 2000] GUNNERSON, Eric: *C#*. Bonn : Galileo Press GmbH, 2000
- [Hans-Dieter Burkhardt 2006] HANS-DIETER BURKHARDT, u.a.: *Informatik bis zum Abitur*. Berlin : PAETEC, 2006
- [Freie und Hansestadt Hamburg 2003] HANSESTADT HAMBURG, Behörde für Bildung und Sport Freie und: *Rahmenplan Wahlpflichtfach Informatik*. 2003
- [Norbert Breier 2003] NORBERT BREIER, Steffen F.: *Informatische Grundbildung*. Berlin : Duden PAETEC, 2003
- [Rosseburg 2007] ROSSEBURG, Kai: *Entwicklung einer Programmierumgebung für Roboterbasierten Informatikunterricht an Schulen*. Bachelorarbeit, HAW Hamburg. 2007
- [Stolt 2001] STOLT, Matthias: *Roboter im Informatikunterricht*. URL [http://users.informatik.haw-hamburg.de/~lego/Projekte/projekte\\_deutsch.html](http://users.informatik.haw-hamburg.de/~lego/Projekte/projekte_deutsch.html). – Zugriffsdatum: 2006-12-27, 2001