



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA  
DCA-DEPARTAMENTO DE COMPUTAÇÃO E AUTOMAÇÃO



# **RoboEduc- Uma ferramenta para programação de Robôs LEGO**

**Renata Pitta Barros**

Orientador: Prof. Dr. Luiz Marcos Garcia Gonçalves

Co-orientador: M.Sc. Samuel Oliveira de Azevedo

**Monografia de Graduação** apresentada ao  
curso de Engenharia de Computação da  
UFRN como parte dos requisitos para obten-  
ção do título de Engenheiro de Computação.

Natal, RN, junho de 2008

# **RoboEduc- Uma ferramenta para programação de Robôs LEGO**

**Renata Pitta Barros**

Monografia de Graduação aprovada em 27 de Junho de 2008 pela banca examinadora composta pelos seguintes membros:

---

Prof. Dr. Luiz Marcos Garcia Gonçalves (orientador) ..... DCA/UFRN

---

Prof. Dr. Samuel Oliveria de Azevedo (co-orientador) ..... DCA/UFRN

---

Prof. Dra. Ana Maria Guimaraes Guerreiro ..... DCA/UFRN

---

Alzira Ferreira da Silva ..... DCA/UFRN

*Dedico este trabalho primeiramente  
a Deus, pois sem Ele, nada seria  
possível e não estaríamos aqui  
reunidos, desfrutando, juntos, destes  
momentos que nos são tão  
importantes. Aos meus pais Carlos e  
Cristina; pelo esforço, dedicação e  
compreensão, em todos os momentos  
desta e de outras caminhadas.*

---

# Agradecimentos

---

Ao meu orientador professor Luiz Marcos Garcia Gonçalves, por acreditar no meu trabalho, sou grata pela orientação.

Ao meu co-orientador Samuel Oliveira de Azevedo, muitíssimo obrigada pelo apoio e ajuda.

Ao meu namorado Jamier pela ajuda na revisão deste trabalho.

Aos meus colegas de laboratório Natalnet-DCA, pelas sugestões e críticas.

As crianças das escolas onde pude mostrar o meu trabalho.

As meninas do meu projeto, pelo trabalho em equipe.

À minha família pelo apoio durante esta jornada.

---

# Resumo

---

Este trabalho propõe a utilização de robôs como ferramentas de apoio ao aprendizado colaborativo de pequenos grupos. Com esse intuito foi desenvolvido um software de robótica educacional, denominado RoboEduc. O software foi implementado com o intuito de promover a inclusão digital, principalmente com crianças carêntes, através da robótica.

Este trabalho explica as principais características deste software. Sua interface foi desenvolvida pensando no público que não possui ou possui pouca familiarização com a tecnologia o que o torna uma ótima ferramenta em ambientes para inclusão digital. Através deste software pode-se ensinar conceitos de robótica e programação de uma forma mais atraente e inovadora.

A proposta é que este sistema possa ser utilizado tanto para controlar um protótipo previamente montado, quanto para programá-lo. Todas as fases do sistema foram totalmente desenvolvidas e estão prontas para a utilização. No que diz respeito à programação foram criados níveis com interfaces e funcionalidades diferentes que serão descritos ao longo deste trabalho. Assim também como a linguagem RoboEduc, com suas estruturas e suas palavras reservadas.

Foram realizados experimentos na Escola Municipal Ascendino de Almeida, localizada na periferia da cidade do Natal, e os resultados obtidos comprovam a real eficácia do sistema RoboEduc. As atividades realizadas foram oficinas de robótica com crianças do ensino fundamental.

**Palavras-chave:** Robótica Educacional, Inclusão Digital, Softwares Educacionais, Robótica.

---

# Sumário

---

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação e Justificativa . . . . .	2
1.2	Escopo do Trabalho . . . . .	2
1.3	Organização desta Monografia . . . . .	3
<b>2</b>	<b>Estado da Arte</b>	<b>5</b>
2.1	As Tecnologias Utilizadas . . . . .	5
2.1.1	O <i>framework</i> Qt-4 . . . . .	5
2.1.2	O Sistema Operacional BrickOS . . . . .	6
2.1.3	Programação Robótica . . . . .	6
2.1.4	Os Kits de Robótica Educacional . . . . .	7
2.2	A Educação e a Robótica Educacional . . . . .	11
2.2.1	A Inclusão Digital . . . . .	11
2.3	Os Softwares Educacionais . . . . .	13
2.3.1	O RoboLab . . . . .	14
2.3.2	O SuperLogo . . . . .	14
<b>3</b>	<b>O Software RoboEduc (Versão 3.0)</b>	<b>17</b>
3.1	Concepção do RoboEduc . . . . .	17
3.2	Metodologia de desenvolvimento de software . . . . .	18
3.2.1	Visões UML . . . . .	18
3.2.2	Diagramas UML: <i>Use-Case</i> . . . . .	19
3.3	Implementação do RoboEduc . . . . .	23
3.3.1	Gramática da Linguagem . . . . .	25
3.3.2	As Palavras Reservadas da Linguagem . . . . .	25
3.3.3	O interpretador do RoboEduc . . . . .	25
3.3.4	Diagramas UML: Classe . . . . .	26
<b>4</b>	<b>Experimentos e Resultados</b>	<b>28</b>
4.1	Experimentos . . . . .	28
4.2	Resultados . . . . .	30
4.3	Usando o RoboEduc versão 3.0 . . . . .	33
<b>5</b>	<b>Conclusões</b>	<b>40</b>
	<b>Referências bibliográficas</b>	<b>41</b>

---

# Lista de Figuras

---

2.1	Kit da Robokit . . . . .	8
2.2	Kit da CyberBox . . . . .	8
2.3	Kit da Alfa . . . . .	9
2.4	Ambiente de Programação Alfa . . . . .	10
3.1	Diagrama de caso de uso do <i>Autor Autoria</i> . . . . .	19
3.2	Diagrama de caso de uso do <i>Autor Aluno</i> . . . . .	20
3.3	Estrutura do arquivo XML . . . . .	24
3.4	As palavras reservadas da linguagem RoboEduc . . . . .	26
3.5	Diagrama das principais classes do RoboEduc . . . . .	27
4.1	Oficinas de Robótica . . . . .	30
4.2	Tela Inicial do RoboEduc . . . . .	33
4.3	Tela para criar modelos, bases, atuadores, sensores, ações, sentidos . . . . .	34
4.4	Tela para Criar Protótipos . . . . .	34
4.5	Selecionar Protótipos Cadastrados . . . . .	35
4.6	Selecionar o que se deseja fazer com o protótipo . . . . .	35
4.7	Tela de controle do protótipo . . . . .	36
4.8	Tela para ensinar o protótipo . . . . .	36
4.9	Tela dos Níveis de Programação . . . . .	37
4.10	Tela do Nível 1 de Programação . . . . .	37
4.11	Tela do Nível 2 de Programação . . . . .	38
4.12	Tela do Nível 3 de Programação . . . . .	38
4.13	Tela do Nível 4 de Programação . . . . .	39
4.14	Tela do Nível 5 de Programação . . . . .	39

---

# Lista de Tabelas

---

2.1	Análise Comparativo dos Softwares de Robótica Pedagógica . . . . .	15
-----	--	----



---

# Capítulo 1

## Introdução

---

As novas tecnologias, em particular o advento do computador, surgem e provocam alterações no comportamento da sociedade, como, um passado recente, aconteceu com a televisão, o rádio e o telefone. Atualmente, somos considerados a sociedade da informação, tendo em nossas mãos uma infinidade de soluções digitais cada vez mais surpreendentes.

No entanto, todos estes avanços ainda não estão disponíveis para toda a sociedade, em virtude dos altos custos, da falta de infra-estrutura, da ausência de professores capacitados e, principalmente, da falta de uma política bem definida para a inclusão digital.

As novas tecnologias de comunicação e informação devem ser compartilhadas o quanto antes, caso contrário corremos o risco de criar uma nova geração: os excluídos digitais. A Sociedade da Informação busca reunir a todos, sua democratização deve possibilitar que toda a população tenha acesso às novas tecnologias, utilizando-as com todo o seu potencial.

Este assunto é de relevância mundial, diversos organismos internacionais como o Banco Mundial, o Fórum Econômico Mundial, o G-8 vêm trabalhando na busca de soluções e no alerta para o perigo do analfabetismo digital. O principal documento que ratifica a importância da Inclusão Digital foi publicado pela Organização das Nações Unidas (ONU) no final do ano passado. Em seu Relatório Anual de Desenvolvimento Humano, a Organização não só priorizou as novas tecnologias como as colocou como facilitadora de avanços sociais.

Para melhor visualizar a questão, a ONU estabeleceu um novo indicador para o Desenvolvimento Humano: o Índice de Avanço Tecnológico (IAT), criado para avaliar a produção e disseminação das novas tecnologias e, acima disso, seu aproveitamento pela população. Foram analisados 72 países onde houve acesso à dados confiáveis. O Brasil ficou em 43º lugar. O índice leva em conta a criação e capacidade de inovação em novas tecnologias, difusão das mais recentes conquistas assim como das tecnologias mais antigas (eletricidade e telefonia, por exemplo) e habilidade intelectual, ou seja, a taxa de escolaridade. A delicada posição do Brasil no ranque, atrás de países como Panamá, Trinidad e Tobago e Romênia, mostra que precisamos investir ainda muito mais tempo e recursos. [ONU 2008]

A Exclusão Digital é uma realidade que assola a sociedade brasileira. A alteração deste quadro acontecerá a partir da mobilização de esforços coordenados de empresas, governo e sociedade civil para a criação de condições à inclusão digital.

Neste sentido, tem se buscado facilitar a aquisição do conhecimento, atendendo assim, as necessidades do mundo atual. Dentro desta perspectiva a robótica educacional representa uma ferramenta que possibilita a sociedade o conhecimento da tecnologia atual de maneira dinâmica, desenvolvendo um grande número de habilidades e competências.

A interação entre robótica e educação vem crescendo ao longo das últimas décadas. Devido a esse crescimento o Departamento de Engenharia de Computação e Automoção (DCA) da Universidade Federal do Rio Grande do Norte, desenvolveu um software Educacional denominado RoboEduc, visando promover a utilização da robótica educativa no ensino fundamental e médio.

Há cerca de 4 anos, um projeto piloto foi implantado na escola Municipal Ascendino de Almeida, situada na periferia do município de Natal, Rio Grande do Norte, com oficinas de robótica. As atividades foram desenvolvidas através da Robótica Educacional, onde foram ensinadas inicialmente as noções de espaço, tempo e velocidade [Barrios-Aranibar et al. 2006].

Dessa experiência, concluiu-se que a implantação da robótica educacional, deve passar pelo projeto, montagem e a manipulações dos robôs, que não é um processo trivial. Porém isso não se torna um empecilho, uma vez que existem no mercado kits de robótica relativamente fáceis de construir e de programar. No desenvolvimento deste projeto, foi utilizado o kit da LEGO MindStorms [LEG 2008] por oferecer uma plataforma simples, pela sua parte didática simples e sua fácil manipulação. O uso desse kit contribuiu significativamente para o desenvolvimento desse projeto. Apesar de existirem no mercado tecnologias mais novas, como os NXT da LEGO e outras com o kit da PNCA [PNC 2008].

## 1.1 Motivação e Justificativa

Nos últimos anos, é notória a evolução quantitativa e qualitativa de projetos desenvolvidos para minimizar a exclusão digital que utilizam a robótica como instrumento de ensino, como por exemplo, o ambiente de programação dos kits LEGO: o RoboLab [LEG 2008] e, o mais recente lançamento, o LabView [LEG 2008], que é uma plataforma de programação baseada em blocos.

O atual estilo de programação desses softwares requerem um certo grau de conhecimento em robótica, pois trata-se de softwares com uma linguagem de programação de nível médio, exigindo conhecimento em física e programação.

Visando diminuir a complexidade de programação desses softwares, principalmente para poder estender conceitos de programação robótica e computação a crianças na faixa etária de 06 a 10 anos de escolas públicas que são os principais excluídos digitais atualmente, desenvolvemos neste projeto o software educacional RoboEduc como parte da nossa metodologia educacional.

## 1.2 Escopo do Trabalho

A Inclusão Digital pode também ser denominada de info-inclusão que é a democratização do acesso às tecnologias da informação, buscando permitir a inserção de todos

na sociedade da informação. A inclusão digital não se restringe apenas ao contato com a tecnologia, mas também, visa simplificar a rotina diária, maximizar o tempo e as suas potencialidades. Um incluso digital não é aquele que apenas utiliza essa nova linguagem, para, por exemplo, trocar e-mails, mas sim o que usufrui dessa inovação para melhorar a sua condição de vida.

No intuito de contribuir para o fim da exclusão digital, desenvolve-se um projeto tecnológico educacional na Universidade Federal do Rio Grande do Norte que integra conceitos tecnológicos com aprendizado colaborativo. Este projeto utiliza como principal ferramenta a robótica educacional [Barrios-Aranibar et al. 2006].

O protótipo inicial do software RoboEduc foi utilizado na alfabetização digital. Em geral, o público alvo do RoboEduc são crianças do primeiro ao quinto ano do ensino público, que tiveram pouco contato com esse tipo de tecnologia.

Convém ressaltar que o presente trabalho faz parte de um projeto maior, que se trata da inserção da Robótica Educacional em escolas da rede pública, desenvolvido no Laboratório Natalnet, onde estão envolvidos vários outros alunos (pelo menos um aluno de doutorado, dois de mestrado, e 4 de graduação). Como contribuições principais da nossa pesquisa e trabalhos desenvolvidos no projeto, destacamos os seguintes tópicos onde pudemos colaborar de forma mais efetiva:

1. Participação de experimentos (oficinas) de interação entre os robôs e as crianças na Escola Municipal Ascendino de Almeida.
2. Implementação de um software para controle e programação de plataformas robóticas que seja facilmente manipulável por crianças do primeiro ao quinto ano escolar. Convém ressaltar que participamos desde a primeira versão do software, hoje totalmente modificada em relação à sua versão original.
3. Desenvolvimento de uma linguagem de programação completa para programação de robôs, incluindo interpretador e compilador, visando efetivar a programação dos robôs pelas crianças.
4. Implementação de uma metodologia (ou técnica) para permitir a operação remota dos robôs.
5. Colaboração em artigos que foram apresentados em conferências nacionais e internacionais do grupo [Silva et al. 2008, Barrios-Aranibar et al. 2006].

Seguimos um plano de trabalho onde especificamos as metodologias a serem adotadas no desenvolvimento deste sistema bem como nossa estratégia de ação. A execução desse trabalho considera o paradigma de engenharia de software denominado prototipagem e baseia-se na produção de protótipos que evoluirão rumo ao produto final (paradigma evolutivo), moldando esta metodologia de acordo com nossas necessidades.

### 1.3 Organização desta Monografia

No Capítulo 1 desta monografia foi descrita uma visão geral deste trabalho, as etapas já realizadas, os objetivos e, por fim, a sua estrutura. O Capítulo 2, é introduzido o conceito de robótica, as estruturas de engenharia de software, as tecnologias utilizadas

e um comparativo entre os trabalhos que estão sendo desenvolvidos paralelamente nesta área. O Capítulo 3 descreve o sistema e todas as ferramentas que foram utilizadas para a sua execução, as visões UML, os principais algoritmos implementados. O Capítulo 4 descreve os experimentos realizados, e os resultados da evolução do aprendizado das crianças. O Capítulo 5, descreve as contribuições e as considerações finais, além dos trabalhos futuros e a conclusão.

---

# Capítulo 2

## Estado da Arte

---

Neste Capítulo, são abordadas as metodologias utilizadas para o desenvolvimento do software proposto neste trabalho. Também são apresentadas as idéias dos principais autores da área de software para Robótica Educativa.

### 2.1 As Tecnologias Utilizadas

Um Ambiente Gráfico (ou GUI, do ingles *guided user interface*) é um software desenvolvido para facilitar e tornar prática a utilização do computador e programação através de representações visuais do sistema operacional ou da linguagem de programação. Pensando nisso, uma das tecnologias utilizadas para o desenvolvimento desse projeto foi a GUI conhecida como Qt-4 [Qt 2008].

#### 2.1.1 O *framework* Qt-4

O Qt é um dos kits de desenvolvimento para a construção de GUIs mais conhecido em C++. Um dos primeiros documentos sobre o *framework* do Qt (estrutura de suporte) foi publicado em maio de 1995. O Qt foi inicialmente desenvolvido por Haavard Nord e Eirik Chambe-Eng (Presidente da Trolltech) [Qt 2008]. Em 1991, Haavard começou a escrever as classes do Qt, com a colaboração do Eirik no design, depois de alguns anos, Eirik teve a ideia de “Sinais e Slots”, que é um dos grandes diferenciais da Tecnologia, pois consite no mecanismo de comunicação entre objetos. Um sinal é emitido por um objeto quando este muda de estado, chamando um *Slot* associado, que são funções executadas no momento do recebimento deste sinal [Qt 2008].

Dentre as vantagens que o Qt oferece, uma delas é a compatibilidade de um único código-fonte, isso permite a sua reutilização. Um vez escrita a aplicação, seria somente re-compilar para se obter uma versão do programa no sistema operacional desejado. Além disso o Qt possui diversas e boas características, ótimo desempenho em C++, e documentação técnica de alta qualidade.

A empresa responsável pela criação do Qt é a norueguesa Trolltech. A companhia foi incorporada em Março de 1994. Originalmente, era chamada de Quasar Technologies, depois, se tornou a Troll Tech e, hoje em dia é chamada de Trolltech [Qt 2008].

O Qt é um sistema multi-plataforma para o desenvolvimento de programas baseados em interface gráfica e utiliza o paradigma de programação orientado a objeto. O QT utiliza C++ com diversas extensões não padronizadas, executadas por um pré-processor adicional capaz de gerar o código padrão de C++ antes da compilação.

Há versões do Qt para diversas outras linguagens de programação, com extensões para Ada, C, Java, Pascal, Perl, PHP (PHP-Qt), ruby (RubyQt), e python (PyQt). O Qt é distribuído pela Trolltech nas seguintes plataformas: Qt/X11, Qt/Mac, Qt/Windows e Qt/Embedded.

A escolha do Qt para a implementação do software RoboEduc, ocorreu pela necessidade de um ambiente de programação gráfica que utilizasse a linguagem C++. Esta necessidade aconteceu devido ao protocolo utilizado para a comunicação, - LNP - [LNP 2008] ser escrito nesta linguagem de programação.

### 2.1.2 O Sistema Operacional BrickOS

Os sistemas operacionais que acompanham os kits com controlador RCX são as versões RCX1 e RCX2, ambas proprietárias da LEGO, de fácil uso, mas com algumas limitações de desempenho e funcionalidade. O trabalho com estes sistemas operacionais ocorre com maior frequência nos compiladores Robolab, RIS ou NQC [Gonçalves et al. 2005].

Como solução para as limitações impostas pelo software que acompanha os kits, existe opção de usar o software de código aberto BrickOs [Bri 2008] que consiste em um sistema operacional (e compilador) onde o usuário possui maior controle sobre o RCX, bem como um uso completo de sua memória. Existe um ambiente para desenvolvimento e execução de programas RCX, baseado na Linguagem C/C++ e mais algumas ferramentas para envio dos programas ao RCX. O BrickOS é um sistema operacional alternativo para o RCX da Lego Mindstorms. Foi desenvolvido originalmente para Linux, mas é possível programar em qualquer editor de texto. Porém, existem alguns programas específicos como o BricxCC (NQC, BrickOS e outros) [Bri 2008] e o NQCEdit (NQC) [NQC 2008] que permitem editar, testar o programa, e carregar o programa no robô via torre infravermelha (num ambiente integrado).

### 2.1.3 Programação Robótica

De acordo com Steffen [STEFFEN 2002], a Robótica pode ser definida como uma área de conhecimento relacionada com o controle e a construção de robôs. O termo robô é uma derivação da palavra tcheca *robotnik* que significa servo. Esta palavra foi utilizada pelo escritor e dramaturgo Karel Capek no ano de 1921 em sua peça teatral de ficção científica intitulada R.U.R. (que significa *Rossum's Universal Robots*), porém, coube ao escritor de ficções científicas e bioquímico famoso, Isaac Asimov a popularização do termo Robô, por meio de mais de 500 publicações envolvendo a temática Robótica. Foi ele também o idealizador das três leis fundamentais da robótica, que dizem:

1. Um robô não pode causar dano a um ser humano nem, por omissão, permitir que um ser humano sofra;

2. Um robô deve obedecer às ordens dadas por seres humanos, exceto quando essas ordens entrarem em conflito com a Primeira Lei;
3. Um robô deve proteger sua própria existência, desde que essa proteção não se choque com a Primeira nem com a Segunda Lei da robótica.

Asimov criou essas leis devido a muitos críticos da época pensarem nos robôs como destruidores da humanidade, viam nessas máquinas um futuro apocalíptico. O escritor, ao contrário, enxergava o robô como um instrumento capaz de auxiliar o homem em determinadas tarefas, possibilitando ao mesmo um maior tempo para atividades mais criativas. [Asimov n.d.].

Atualmente, podemos conceber o robô, seguindo a idéia de Isaac Asimov, como sendo um instrumento capaz de ajudar o homem na realização de tarefas do cotidiano, viabilizando um tempo maior para o seu lazer, para o convívio familiar, para o trabalho intelectual, ou até mesmo para o auxílio em atividades de grandes periculosidades.

A Robótica está muito próxima da vida dos seres humanos, nos eletrodomésticos, nos aparelhos eletrônicos, nos elevadores, em caixas eletrônicos, enfim, em uma infinidade de sistemas. A robótica está presente também no meio industrial, evoluindo a cada dia, principalmente no caso de situações adversas ou de perigo, onde a utilização de um ser humano poderia ser perigosa. Como exemplo, podemos citar as montadoras de automóveis que na suas linhas de montagem, utilizam a robótica para realizar alguns serviços. [ZILLI 2004].

De acordo com Steffen apud D'Abreu [STEFFEN 2002], na educação, a robótica industrial pode estender um conjunto de conceitos básicos em um contexto onde as atividades de construção e controle de dispositivos, usando kits de montar ou outros materiais, como sucata, compostas por diferentes peças, motores e sensores controláveis por computador e softwares, propiciando o manuseio destes conceitos em um ambiente de aprendizagem. Portanto, a robótica aplicada à educação designa-se Robótica Educativa ou Robótica Pedagógica, ambas com o mesmo sentido.

### 2.1.4 Os Kits de Robótica Educacional

Com a evolução da tecnologia, principalmente em hardware e software, vários kits de robótica educacional surgiram no mercado, visando facilitar a criação de protótipos para educação. Alguns desses kits são geralmente usados em cursos de mecatrônica e de engenharia mecânica, visando levar o aluno a um conjunto operacional com mais facilidade, por exemplo simulando uma linha de montagem.

#### Robokit

O Robokit consiste em um kit de robótica educacional que contém basicamente uma caixa controladora independente (funciona sem computador), motores pequenos e *leds* (pequenas luzes). A caixa apresenta um teclado colorido, próprio para crianças e adolescentes. Por meio de programação, nesse teclado é possível controlar os motores e os *leds*. O kit tem por finalidade o desenvolvimento de programação e robótica para alunos da educação básica. O kit é composto por quatro *leds*, dois motores contínuos e um motor







não necessita de computadores potentes. Funciona em todos os sistemas operacionais com acesso à porta serial incluindo: Windows, Linux, Solaris, FreeBSD, etc. [Cyb 2008]

### **Kit da Alfa Hobby 2008**

A empresa ALFA tem diversos kits de robótica, dentre eles a empresa apresenta o Kit ALFA Hobby e o kit ALFA Educ 2008 [PNC 2008].



Figura 2.3: Kit da Alfa

Esse kit é focado para iniciantes na robótica, com esse kit é possível construir e programar robôs. Este kit é formado por: um módulo de controle MC2.5, o programa LEGAL 2008, dois sensores de faixa, dois sensores de contato, um cabo USB, dois motores com redução, duas bases para motores, um par de rodas (75 mm), uma roda livre, peças estruturais metálicas para montagem, porcas e parafusos diversos. Para utilizar esse kit é necessário no mínimo o sistema operacional Windows 2000, XP; CPU Pentium II 450 MHz; 128MB de memória RAM, 30 MB de espaço em disco rígido; 4X de velocidade do CD-ROM; placa de vídeo SVGA compatível com Direct X e uma porta serial disponível.

### **Kit ALFA EDUC 2008**

Esse kit é destinado à instituições de ensino. Desenvolvido para iniciar o trabalho na robótica e na mecatrônica, com ele o usuário poderá projetar, construir e programar robôs e dispositivos mecatrônicos. Este kit é formado por: um módulo de controle MC2.5, um programa LEGAL 2008, dois sensores de luz, dois sensores de contato, dois sensores de faixa, um sensor de temperatura, um sensor de cor, um sensor de infravermelho, um cabo USB, dois motores com redução, duas bases para os motores, rodas de diferentes tamanhos, uma roda livre, um servo motor, peças estruturais metálicas para montagem, porcas e parafusos diversos. Os requisitos do produto são: Sistema operacional Windows 2000, XP; CPU Pentium II 450 MHz; 128 MB de memória RAM, 30 MB de espaço em disco rígido; 4X de velocidade do CD-ROM; placa de vídeo SVGA compatível com Direct X e uma porta serial disponível. O ambiente de programação LEGAL é apresentado na Figura 2.4.

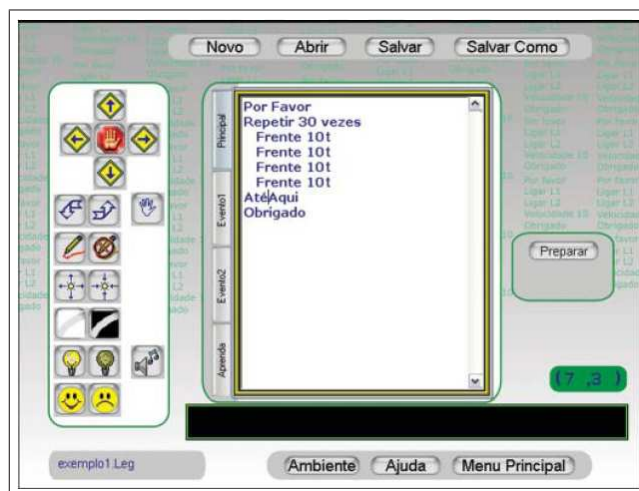


Figura 2.4: Ambiente de Programação Alfa

Todas as ações que o robô deve executar serão definidas pelo usuário nesse ambiente. Os programas são escritos utilizando a linguagem LEGAL. O programa, uma vez escrito e compilado, será descarregado no módulo de controle do robô via cabo serial. Ao término deste processo, o robô estará pronto para funcionar de forma autônoma e poderá ser desconectado do computador principal.

Porém, esse kit de programação é limitado ao número de sensores que contém, pois tem uma programação orientada a eventos. Um sinal capturado por um sensor dispara um desses eventos, porém não é possível programar uma função que interaja com vários sensores ao mesmo tempo.

## LEGO

A tecnologia conhecida como LEGO Mindstorms é uma linha de kits, lançada comercialmente em 1998, voltada para a educação tecnológica. É constituído por um conjunto de peças tijolos cheios, placas, rodas, tijolos vazados, motores, eixos, engrenagens, polias e correntes, acrescido de sensores de toque, de intensidade luminosa e de temperatura, controlados por um processador programável - o módulo RCX (*Robotic Command Explorer*). O conjunto permite criar robôs simples, passíveis de executar funções básicas pré-programadas.

O módulo RCX processa comandos pré-programados em um computador, através de softwares específicos, como o RoboLAB (na versão educativa) ou o Robotics Invention System (na versão comercial), permitindo a interação da estrutura construída com o ambiente no qual se inscreve. O conjunto é fornecido com uma torre baseada em tecnologia de raios infravermelhos, que pode ser conectada a um computador pessoal de duas formas, através da porta USB ou através da porta serial. Estes conjuntos são utilizados para, além da função lúdica, como função didática abordando a teoria e a prática de conteúdos direcionados para a introdução a robótica, permitindo o desenvolvimentos de projetos de pequeno e médio porte, estimulando a criatividade e a solução de problemas do cotidiano

por parte dos alunos. Dentre os kits de robótica educativa, o nosso projeto escolheu o kit da LEGO para construção dos robôs.[LEG 2008]

## 2.2 A Educação e a Robótica Educacional

Segundo o dicionário interativo da educação brasileira [Edu 2008], a robótica educacional é um termo utilizado para caracterizar ambientes de aprendizagem que reúnem materiais de sucata ou kits de montagem compostos por peças diversas, motores e sensores controláveis por computador e softwares que permitam programar de alguma forma o funcionamento dos modelos montados. Muito se fala sobre a contribuição que a robótica pode trazer para ambientes não acadêmicos, como, salas de aula de ensino médio ou, principalmente, para estudantes de escolas primárias. Como uma confirmação desse fato, cada vez mais o uso da robótica está se estendendo para fora do meio acadêmico e se tornando um meio de impulsionar e aprimorar o conhecimento adquirido pelas crianças no tocante as habilidades em matemáticas, conhecimentos artísticos, criatividade, linguagem e, para crianças mais velhas, noções de programação entre várias outras coisas.

O trabalho em ambientes de robótica pedagógica tem vários objetivos e metodologias, apontando, em geral, para que o aluno siga instruções e manuais ou crie e experimente a partir dos materiais específicos desse ambiente.

No mundo inteiro existem projetos de robótica educacional, dentre eles podemos citar o projeto desenvolvido em conjunto pelas Universidades de Columbia e de Cambridge. Este projeto aconteceu no verão de 2003 com aproximadamente cinquenta crianças do segundo grau. O projeto foi desenvolvido dentro de outros dois projetos maiores: Science and Technology Entry Program (STEP) e Playing2Win (P2W). O objetivo principal foi aumentar o conhecimento de disciplinas como ciência e matemática através do desenvolvimento de protótipos utilizando os kits da tecnologia LEGO Mindstorms Robotics Invention System. O público alvo deste projeto foram moradores do bairro de Central Harlem vizinhos a Manhattan, onde 51% da população possuíam rendimento familiar entre baixo e moderado, ou seja, pode ser considerados de classe média baixa. Dentre os estudantes que participaram deste projeto, 67% eram afro-descendentes e 20% possuíam descendência hispânica. O software utilizado foi o Robolab, desenvolvido pela National Instruments and Tufts University, com base no LabView. O Robolab é um software que usa a linguagem de programação gráfica LabVIEW, ele requer certo conhecimento e compreensão das funcionalidades de baixo nível do robô (motores, sensores, etc.).

### 2.2.1 A Inclusão Digital

Quando se fala em inclusão, as pessoas se referem à diminuição da quantidade de excluídos. Uma pessoa excluída é aquela que não tem a capacidade de participar totalmente nas esferas sociais e econômicas devido a uma grande variedade de problemas sociais; incluindo desemprego, falta de instrução, deficiências, dentre outros [Fitch 2002].

Já, quando falamos explicitamente em inclusão digital, a idéia é tentar disponibilizar a todos as oportunidades de utilizarmos os recursos que os computadores, a tecnologia de

informação e a comunicação oferecem. Porém, nem sempre esses recursos estão disponíveis para quem tem algum tipo de “carência”. A ausência de espaço físico, programas adequados e até mesmo, a ausência de monitores capacitados, são exemplos, dentre outras deficiências, que contribuem gradativamente para os chamados “excluídos digitais”. É uma situação tão alarmante que a exclusão digital seria uma mera decorrência da exclusão social. Por toda parte, assistiu-se a iniciativas direcionadas ao combate a esse tipo de exclusão, através do antídoto “inclusão digital”. A tecnologia da informação tem influenciado, de forma tão significativa, que o acesso à informação é um grande passo para combater a exclusão social.

Grande parte das pessoas “digitalmente excluídas” tem um certo receio ou, até em alguns casos, certa “aversão” à tecnologia, principalmente as mais idosas. Porém, segundo a pesquisa feita pela Fundação Getúlio Vargas em 2003, sobre o mapa da exclusão digital no Brasil [Bagio 2003], são nas periferias das cidades onde se concentram a maioria das pessoas excluídas, onde, crianças entre primeira e quarta séries do Ensino Fundamental, possuem ainda menos acesso a tais tecnologias se comparadas aos alunos de quinta à oitava séries ou adolescentes do Ensino Médio. Além disso, as crianças que não têm acesso aos computadores possuem um menor desempenho em matemática em comparação aquelas que possuem um ou mais computadores [Barrios-Aranibar et al. 2006]. Existe uma grande preocupação para diminuir a exclusão digital no mundo [Fitch 2002, Meer 2003, Wilson et al. 2003, Selwyn 2004, Willis & Tranter 2006] e, é claro, no Brasil não é diferente [da Silva Filho 2003, Rondelli 2003, Porcaro 2006]. Neste país, projetos envolvendo inclusão digital estão fortemente direcionados a “usar computadores e internet”. Estes tipos de projetos são muito importantes para acelerar o processo de introdução da cultura digital, mostrando as pessoas outras formas de interagir com o mundo. Nós cogitamos que os robôs têm um grande potencial de serem aceitos pelas pessoas, em comparação com computadores. Existem várias razões para fazer esta afirmação. Por exemplo, em robótica, uma forma de testar os conhecimentos aprendidos e outras habilidades complexas adquiridas pelos pesquisadores é a utilização de competições que em geral são acolhidas facilmente pelas pessoas. Um outro exemplo é que, para uma pessoa aprender a utilizar um computador, ela precisaria, na maioria dos casos, assistir um curso tradicional; porém, ao usarmos robôs, este aprendizado pode ser mais interativo.

Robôs podem ser úteis no treinamento de habilidades básicas em computação, como operação de computadores usando comandos de texto, comandos visuais, programação, redes etc. Robôs podem ser úteis também quando quer se ensinar matemática ou física, por exemplo, com um robô é possível ensinar noções de distância, velocidade etc.

Robôs não são tão simples de desenvolver, então é necessário simplificar este contexto. Felizmente, existem plataformas simples de robótica, relativamente fáceis de construir e programar. No corrente trabalho, nós usamos kits Lego Mindstorms, uma tecnologia usada geralmente em cursos básicos de robótica. Eles são simples de manipular e, também, são didáticos. Então, eles se tornaram uma boa opção para ensinar robótica as pessoas que estão iniciando o aprendizado na área tecnológica.

Dessa forma, sua utilização nos auxilia no propósito de atingirmos nossos objetivos.

## 2.3 Os Softwares Educacionais

É notável a grande presença de softwares educacionais nas escolas, porém, a falta de filtros e ferramentas que possibilitem uma análise da qualidade destes sistemas, favorece a proliferação no mercado de produtos cujo potencial educativo suscita uma grande interrogação. A dificuldade em se implementar softwares educativos de boa qualidade já vem sendo alvo de pesquisas a algum tempo. Alguns autores já mencionavam a complexidade existente na produção deste tipo de sistema.[D.W.Carraher 1990]

O desenvolvimento de tais sistemas deve ser uma tarefa conjunta entre programadores, designers e professores, porém, a interação entre esses profissionais se torna problemática devido às dificuldades de se compartilhar conceitos de áreas diferentes.

Ao longo desta pesquisa, verificou-se quase a inexistência de empresas brasileiras especializadas na implementação de softwares educacionais. As empresas citadas abaixo são especializadas na distribuição de softwares e na assessoria da área pedagógica de softwares educativos.

- Re-Criar Assessoria e Desenvolvimento de Tecnologia Educacional: a empresa mantém um site informática aplicada à educação e é dedicada exclusivamente a análise de softwares. O site traz análises sob a perspectiva de uso educacional e comentários do ponto de vista técnico, informando a configuração mínima exigida pelo software.[Re- 2008]
- RCTSoft Softwares Educacionais: empresa brasileira que desenvolve softwares educacionais desde 1995. [RCT 2008]
- SoftMarket: empresa que fornece Softwares Educativos de diferentes fabricantes para as Escolas. [Sof 2008]
- Informar Educacional: empresa que atua, exclusivamente, na área de informática educacional. Criada em 1988, trabalha prestando assessoria a educandários e colocando o computador como uma ferramenta no processo ensino aprendizagem. [Inf 2008a]

Foi encontrada uma organização que desenvolve softwares livres para a educação, a Tux4Kids [Inf 2008b]. Existem vários softwares mantidos e em desenvolvimento em sua web page.

As empresas citadas anteriormente possuem catálogos via internet para escolha e visualização de alguns dos seus produtos. Em geral, os softwares encontrados são extremamente direcionados a uma determinada disciplina. No contexto de utilização de tais ferramentas não se verificou a existência de uma característica muito importante no que diz respeito à educação, - a interdisciplinariedade -, característica que naturalmente está presente em um ambiente de robótica educacional.

Já há alguns softwares que utilizam a robótica com fins pedagógicos, neste contexto, podemos citar o RoboLab e o SuperLogo . O RoboLab é um software desenvolvido pela National Instruments and Tufts University, com base no LabView. Já o SuperLogo é uma versão do Logo desenvolvida pela Universidade de Berkley (EUA) traduzida para o português pelo Núcleo de Informática Aplicada a Educação (NIED) da UNICAMP.



### 2.3.1 O RoboLab

Esse é um ambiente de programação que vem junto aos kit's da linha LEGO Minds-torms, esse software está na versão 2.5, seus desenvolvedores disponibilizam tutoriais para fixação dos ícones utilizados pelo programa dando suporte para o usuário que utiliza a tecnologia LEGO pela primeira vez. Existe um breve guia que ajuda a instalar o programa, ensina os princípios básicos de programação e contém as funções que podem ser utilizadas, porém, é muito limitado. Os programas feitos no RoboLab podem ser passados para o RCX através de uma torre (IR) infravermelha. A torre IR está disponível com uma conexão serial (porta COM) ou para uma conexão USB (para 2.5 ou versões mais elevadas somente). Requer sistemas como o Windows 95 / 98 / 2000 / XP, Processador 486/33 MHz, 32 MB RAM, 70 MB de HD, e uma porta serial. Os Macintosh são sistemas a partir de 7.5 com 60 MHz PowerMac 2122 Processador, 32 MB RAM, 85 MB HD, e uma porta serial. O programa Robolab é dividido em três níveis de programação e permite que o nível de programação combine o conhecimento e as habilidades do estudante, como o Piloto, Inventor e Investigador. Em todas essas etapas, a programação é realizada através de ícones que é uma linguagem de programação gráfica LabVIEW. Ele requer certo conhecimento e compreensão das funcionalidades de baixo nível do robô (motores, sensores, etc.). É ideal para projetos de nível médio ou elementar.

### 2.3.2 O SuperLogo

O SuperLogo é uma versão do Logo desenvolvida, em meados dos anos 60, pelo professor Seymour Papert e outros pesquisadores do Massachusetts Institute of Technology (MIT). Essa linguagem apresenta um grande grau de flexibilidade, podendo ser utilizada tanto por crianças como por programadores experientes, atendendo, em ambos os casos, as necessidades do usuário. Tem-se uma terminologia simplificada. Entende-se por terminologia simplificada a facilidade no que diz respeito a termos de nomes de comandos, de regras sintáticas e de uma parte gráfica. Um ponto bastante interessante no Logo e, conseqüentemente, no SuperLogo é a parte gráfica caracterizada pela presença de um cursor representado pela figura de uma Tartaruga que pode ser deslocada no espaço da tela através de alguns comandos relacionados ao deslocamento e giro da mesma. A realização de movimentos da tartaruga na janela gráfica utiliza comandos de deslocamento e giro, como por exemplo: *para frente (ou pf)*, *para trás (ou pt)*, *para direita (ou pd)* e *para esquerda (ou pe)*. Para usar estes comandos, são necessários especificar o número de passos ou o grau do giro. Por exemplo: *pf 80 (andar para frente 80 passos)*, *pd 90 (virar à direita 90)*, *pt 30 (andar para trás 30 passos)* e *pe 15 (virar à esquerda 15)*.

Após a análise desses dois softwares, podemos, constatar que são ótimas ferramentas para serem utilizadas com pessoas que tenham um embasamento teórico sobre robótica, programação e informática. Porém, essa não é realidade da maioria das pessoas de baixa renda, assim, esses sistemas não são adequados para serem utilizados como um instrumento para inclusão digital. Suas interfaces, embora sejam através de ícones, são bastante difíceis de serem entendidas, fazendo com que os usuários não utilizem todos os recursos disponíveis no sistema.

Na Tabela 2.1, incluímos um item chamado nível de complexidade (ou também en-

Tabela 2.1: Análise Comparativo dos Softwares de Robótica Pedagógica

	BrickOS	NQC	LejOS	RoboLab	SuperLogo
<b>Linguagem Base</b>	C	C	Java	LabView	Logo
<b>Entrada de Dados</b>	texto	texto	texto	visual	texto
<b>Necessário Conhecer:</b>					
<i>Programação</i>	Sim	Sim	Sim	Não	Sim
<i>Robótica</i>	Sim	Sim	Sim	Sim	Não
<i>Escrita</i>	Sim	Sim	Sim	Não	Sim
<b>Utiliza Conceitos de:</b>					
<i>Motores</i>	Sim	Sim	Sim	Sim	Não
<i>Sensores</i>	Sim	Sim	Sim	Sim	Não
<i>Programação Básica</i>	Sim	Sim	Sim	Não	Sim
<i>Controle de Fluxo</i>	Sim	Sim	Sim	Sim	Sim
<i>Programação Avançada</i>	Sim	Sim	Sim	Não	Não
<b>Nível de complexidade</b>	5	5	5	4	3

tendido como nível de abstração). Os valores deste item foram colocados considerando a existência de um total de cinco níveis de abstração. Esta divisão em cinco níveis é uma outra proposta do nosso grupo, foi formulada após as análises procedidas sobre os softwares existentes para programação robótica, citados na Tabela 2.1. Os dois primeiros níveis possuem a complexidade mínima para o usuário, ou seja, nesses níveis não é trabalhado conceitos de motores, sensores nem conceitos de programação avançada. O primeiro nível considera a programação do robô de forma simples, utilizando elementos visuais que representam as funcionalidades do mesmo, como por exemplo *andar para a frente, andar de ré, virar à direita ou esquerda, fechar a garra*, etc.

Neste primeiro nível, as ações dos robôs são inseridas pelo programador em forma sequencial. Já o segundo nível acrescenta o conceito de controle de fluxo, mas mantém a interface gráfica. Desta forma, o estudante se prepara para o terceiro nível, onde os mecanismos são basicamente semelhantes, no entanto, utiliza comandos de linha (escritos pelo teclado em um programa textual, ou script). A não consideração da linguagem escrita nesses níveis iniciais representa uma importante característica devido possibilidade das pessoas excluídas digitalmente ser também vítimas de exclusão social, ou seja, existe sempre a possibilidade de depararmos com alunos analfabetos, que poderiam se sentir mal perante tal situação.

Convém ressaltar que o software proposto nesta monografia incluirá todos os cinco níveis de programação, além de incluir um nível zero de complexidade, onde o robô não é programado, e sim controlado remotamente. Desta forma, o estudante leigo ou a pessoa excluída digitalmente terá as condições de assimilar estes conhecimentos.

Através das pesquisas vimos que existem várias ferramentas (livres e comerciais) para programação de robôs. Todas elas são voltadas a usuários com certos conhecimentos em computação e com pelo menos conceitos básicos de robótica. Por exemplo, quando

falamos de linguagens de programação para LEGO Mindstorms, a linguagem brickOS [Bri 2008] e o NQC [NQC 2008] são baseados na linguagem de programação C e não são simples para as crianças entenderem, principalmente se elas não possuem intimidade com a tecnologia. O mesmo acontece com o programa leJOS, [Lej 2008] baseado na linguagem de programação Java.

É notável a dificuldade de compreensão de muitos conceitos para as crianças. Então, é necessário um novo nível de abstração que forneça uma maneira das crianças entenderem melhor as funcionalidades de baixo nível dos robôs e suas respectivas partes, tais como motores, sensores, etc. Além disso, é fundamental que esse novo nível de abstração envolva conceitos de programação. Todos os projetos encontrados no estudo do estado da arte requerem que os usuários conheçam conceitos de robótica, física ou linguagem de programação. É por este motivo que, para esta monografia, estamos desenvolvendo um programa para ensinar robótica a crianças que será explanado no Capítulo 3.



---

## Capítulo 3

# O Software RoboEduc (Versão 3.0)

---

Com o intuito de utilizar a robótica educacional como ferramenta para a inclusão digital, principalmente para alunos iniciantes na vida escolar, foi realizada uma análise sobre as ferramentas disponíveis para tal propósito. Concluímos que estas ferramentas não favoreciam a inclusão digital, portanto, tornou-se necessário à criação de um software de robótica educacional específico que permitisse o ensino da robótica utilizando técnicas de aprendizado colaborativo desde o nível mais básico. O software proposto denomina-se RoboEduc.

### 3.1 Concepção do RoboEduc

O RoboEduc foi inicialmente concebido para ser um software de Robótica Educacional destinado a programação de micro-controladores que utilizam a linguagem de programação BrickOS [Bri 2008]. Atualmente, este conceito evoluiu, sendo que temos intenção de que ele seja usado com qualquer kit de robótica, desde que o kit tenha um conjunto de instruções de baixo nível condizentes com a atual implementação. No caso, basta trocar o conjunto de instruções de baixo nível (de manipulação dos robôs) em nossa ferramenta, para que se tenha a mesma funcionalidade. A implementação disso é elementar, qualquer que seja o kit usado. O RoboEduc foi concebido para um amplo público, pois é destinado a alunos desde a educação infantil até estudantes universitários, mas estamos trabalhando atualmente nas implementações voltadas ao ensino fundamental.

A ferramenta desenvolvida tem o propósito de ensinar robótica, linguagem de programação, lógica de algoritmos e informática, além de poder ser usada em outros conteúdos transversais (Português, História, Ciências, Física, Matemática etc). Ainda, o RoboEduc caracteriza-se por ser uma ferramenta multi-plataforma, ou seja, é possível ser executado em mais de um sistema operacional, tratando-se de um software livre e com sua licença de uso gratuita.

A partir deste software, é possível que os alunos possam projetar o robô, através da criação de modelos, bases, atuadores, sensores e ações. Após, é possível construir um protótipo, delegando um conjunto tarefas (jogos educativos), controlando, ensinando e programando em cinco níveis de complexidades diferentes.

A interação com o usuário ocorre por meio de uma interface simples, estimulante e atrativa, com muitas cores e desenhos, que, na realidade, consiste de artifícios para pro-

porcionar ao usuário o aprendizado de conceitos complexos, como por exemplo, técnicas de programação. O usuário pode acessar o sistema de duas maneiras: Como professor, que pode acessar todas as funcionalidades, e como aluno, onde o seu acesso é restrito. As funcionalidades podem ser executadas através do teclado ou do *mouse*, proporcionando uma familiarização com esses dois tipos de periféricos. A comunicação entre o software e o robô é realizada por uma torre infravermelha conectada ao computador pela porta USB, e o protocolo de comunicação utilizado é o LNP.

Para o desenvolvimento da ferramenta foi realizado no ambiente de programação C++ e as interfaces com o usuário foram programadas com a biblioteca Qt 4.3. Os arquivos de conteúdo estão no formato XML e os arquivos para a manipulação dos robôs foram programados na linguagem C, tendo como biblioteca de interface com o hardware o BrickOS. As ferramentas utilizadas para a implementação do software foram o compilador GCC, a biblioteca gráfica Qt 4.3, o compilador BrickOS e o Eclipse, que é uma IDE (*Integrated Development Environment*) que reúne características e ferramentas de apoio ao desenvolvimento de um software.

## 3.2 Metodologia de desenvolvimento de software

A metodologia utilizada no desenvolvimento do RoboEduc é a Prototipagem Evolutiva. Esta metodologia caracteriza-se por ser baseada numa visão evolutiva do desenvolvimento do software. Esta abordagem envolve a produção de versões iniciais do sistema futuro com o qual pode-se realizar verificações e experimentações para se avaliar algumas de suas qualidades antes que o sistema venha realmente a ser constituído. À medida que o sistema vai sendo desenvolvido, diversas funcionalidades são adicionadas ao protótipo e traduzidas numa eficiente implementação.

O uso desta metodologia no processo de análise e síntese na Engenharia de Software é cada vez mais utilizado. A prototipagem facilita a comunicação entre o cliente e o desenvolvedor. Esse método, focado na criação de protótipos, não, significa que não se tenha uma escrita formal dos requisitos do sistema, uma vez que é necessário estabelecer uma definição clara do projeto.

O software descrito nesta monografia está no seu terceiro protótipo. Os primeiros protótipos estão descritos em documentos anteriores, objetos de monografias e dissertações [dos Santos 2006, Trindade 2007].

Neste terceiro protótipo acrescentamos o **módulo Autoria**, com suas respectivas funcionalidades, e o módulo de programação com os cinco níveis de complexidade que estão descritos no casos de uso, vistos adiante.

### 3.2.1 Visões UML

As visões UML são utilizadas para descrever os diferentes aspectos do sistema, representa uma abstração formada por um conjunto de diagramas. A partir de um conjunto de visões é possível realizar uma descrição completa do sistema a ser constituído.

As visões elaboradas para o RoboEduc foram:

1. Visão de Casos de Uso: mostra a funcionalidade do sistema do ponto de vista externo, é representado por um conjunto de atores que interagem com o sistema e descrito pelos diagramas de casos de uso.
2. Visão Lógica: Descrevem a organização do sistema, os módulos principais, os relacionamentos entre os módulos e suas funcionalidades. A visão Lógica envolve a estrutura estática do sistema (módulos, casses, objetos, relacionamentos) representada pelo Diagrama de Classes.

### 3.2.2 Diagramas UML:Use-Case

As funcionalidades do software podem ser visualizadas nos diagramas mostrados nas Figuras 3.1 e 3.2. Nessas Figuras, é possível observar as ações que o usuário pode executar no sistema. Cada *Caso de Uso* será descrito a seguir.

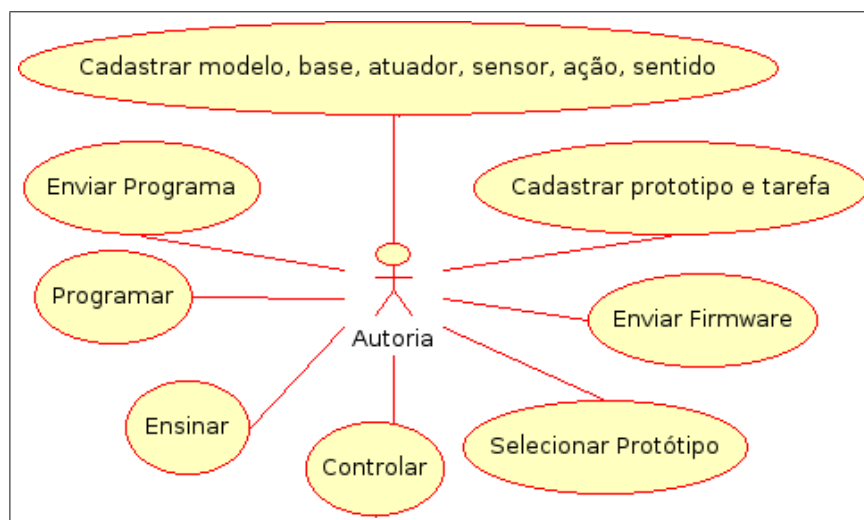
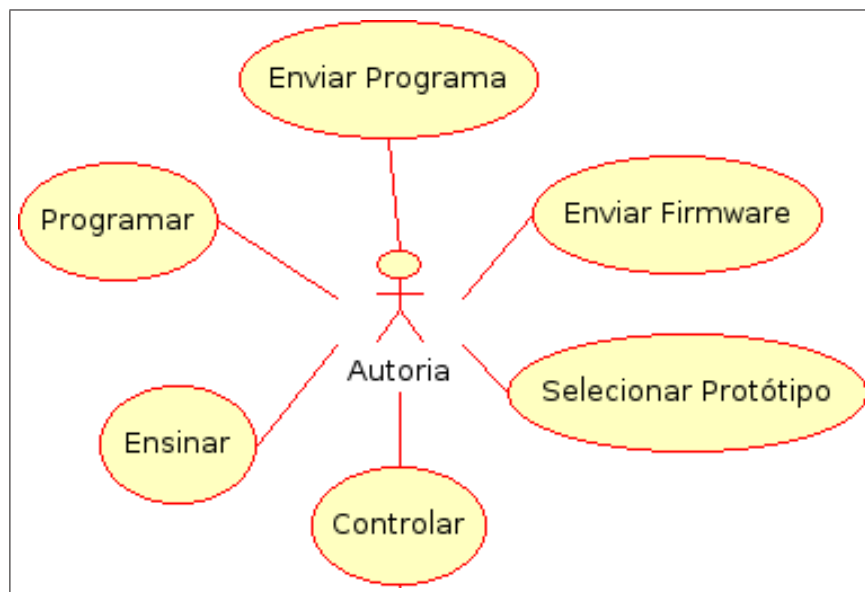


Figura 3.1: Diagrama de caso de uso do *Autor Autoria*

#### Selecionar Módulo Aluno ou Módulo Autoria

Na tela principal da ferramenta RoboEduc, é disponibilizado ao usuário o módulo *Aluno* e o módulo *Autoria*. Essas funcionalidades estão descritas em seguida. No módulo *Autoria* o usuário pode:

1. *Cadastrar modelo, base, atuador, sensor, ação, sentido;*
2. *Cadastrar protótipos e tarefas;*
3. *Enviar Firmware para o robô;*
4. *Selecionar protótipos cadastrados;*
5. *Controlar protótipo;*
6. *Ensinar protótipo;*
7. *Programar protótipo;*

Figura 3.2: Diagrama de caso de uso do *Autor Aluno*

8. *Compilar programas;*
9. *Enviar programas.*

No módulo *Aluno* o usuário pode:

1. *Enviar Firmware para o robô;*
2. *Selecionar protótipos cadastrados;*
3. *Controlar protótipo;*
4. *Ensinar protótipo;*
5. *Programar protótipo;*
6. *Compilar programas;*
7. *Enviar programas.*

### Enviar Firmware

Neste caso de uso, o protocolo de comunicação, entre o robô e o computador, é iniciado quando o usuário clica no botão *Firmware*. O *firmware* é um Sistema Operacional que possui uma lista de prioridades para alocação de tarefas (*preemptive*), que controla diretamente o hardware. A comunicação é realizada através do protocolo LNP (protocolo de troca de mensagens presente no núcleo (*kernel*) do Sistema Operacional do BRICKOS). A transmissão física ocorre usando uma torre de comunicação infravermelha conectada ao computador pela porta USB.

### Cadastrar modelo, base, atuador, sensor, ação e sentido

O caso de uso *Cadastrar Modelo, Base, Atuador, Sensor, Ação e Sentido*, apresenta a tela na qual são exibidos os modelos já cadastrados. O usuário *Austria*, ao clicar no botão

*Criar*, pode cadastrar o nome, a descrição e associar uma imagem do tipo *gif animado* ao novo modelo. Isso é realizado também para Bases, Atuadores e Sensores. Na criação das Ações e dos Sentidos, além dos itens anteriores, também é cadastrado o conteúdo das ações e dos sentidos. Em ambas, existem nas telas de criação a opção de construir o manual de cada item novo.

### **Cadastrar protótipos e tarefas**

Neste caso de Uso, o usuário *Autoria* seleciona um modelo, uma base, um atuador, um sensor, uma ou mais ações, um ou mais sentidos, cadastra um nome para o protótipo, cadastra uma descrição em linhas gerias e associa uma imagem do tipo *gif animado* para o protótipo. Existe também, o cadastro de tarefas que são associadas ao protótipo que esta sendo montado. Após escolher todas as opções, o usuário clica no botão *Protótipo* e o sistema salva um arquivo com as informações referentes ao protótipo criado.

### **Selecionar protótipos cadastrados**

Neste caso de uso, o sistema disponibiliza ao usuário (*Aluno e Autoria*) os modelos de robôs que foram previamente cadastrados no caso de uso *Cadastrar Modelo, Base, Atuador, Sensor, Ação e Sentido*. Após escolher um modelo, são apresentadas ao usuário as opções de construção para o determinado protótipo cadastrado pelo professor e, então, o usuário escolhe os componentes existentes no protótipo montado. Após esta etapa, será apresentada ao usuário uma tela com a descrição, a tarefa e uma imagem do tipo *gif animado* do protótipo escolhido. A próxima etapa é enviar o programa de escuta ao robô, sendo estabelecida a comunicação entre robô e o computador.

### **Controlar protótipo**

No caso de uso *Controlar Protótipo*, as ações que esse protótipo pode executar são exibidas. Será inicializado o protocolo de comunicação entre o robô e o computador. Cada opção representa uma ação que foi previamente cadastrada para o protótipo selecionado. Esses comandos são enviados através do protocolo de comunicação ao robô.

### **Ensinar protótipo**

Este caso de uso aborda, além das funcionalidades descritas no caso de uso *Controlar Protótipo*, uma outra opção é para armazenar os comandos executados em um arquivo e torna-lo apto a ser utilizado posteriormente. Uma outra funcionalidade deste caso de uso é abrir um arquivo existente e executá-lo. A tradução dos comandos e o envio do programa ao robô são feitos internamente pelo sistema, de forma transparente ao usuário.

### **Programar protótipo**

Neste caso de uso, uma tela com os cinco níveis de programação possíveis do RoboEduc é exibida ao usuário. O nível de programação refere-se ao nível de abstração dos objetos disponíveis para a programação do robô. Os níveis de programação são:

- **Nível 1:** As funções são semelhantes ao caso de uso *Controlar Protótipo*. A programação é realizada no modo gráfico por meio do mecanismo *DragDrop*. As funcionalidades deste nível são: *Compilar programa*, *Enviar programa*, *Salvar programa*, *Abrir programa*. A tradução da linguagem gráfica para a linguagem de BrickOS é transparente ao usuário.
- **Nível 2:** Neste nível as funções são semelhantes ao nível anterior. São acrescentadas as estruturas de fluxos. A programação neste nível também é realizada de modo gráfico por meio do mecanismo *DragDrop*. As funcionalidades são: *Compilar programa*, *Enviar programa*, *Salvar programa* e *Abrir programa*. A tradução da linguagem gráfica para a linguagem de BrickOS é transparente ao usuário.
- **Nível 3:** Este nível tem as funções semelhantes ao nível 2, porém a programação é realizado no modo texto. A linguagem de programação utilizada é “RoboEduc”, que foi desenvolvida em conjunto com o software RoboEduc. A interface com o usuário é um editor de texto semelhante aos editores de outras linguagens existentes. A tradução da linguagem RoboEduc para a linguagem de BrickOS é transparente ao usuário. As funcionalidades são: *Utilidades do editor de texto*, *Compilar programa*, *Enviar programa*, *Salvar programa*, *Abrir programa*.
- **Nível 4:** Neste nível, as funções são semelhantes aos do RoboLab, sua programação é realizaa em modo gráfico por meio do mecanismo *DragDrop*. A tradução da linguagem gráfica para o BrickOS é transparente para o usuário. As funcionalidades são: *Compilar programa*, *Enviar programa*, *Salvar programa*, *Abrir programa*.
- **Nível 5:** Neste nível as funcionalidades são as funções do BrickOS. A programação é realizada no modo texto. A interface com o usuário é um editor de texto semelhante aos editores de outras linguagens existentes. As funcionalidades são: *Compilar programa*, *Enviar programa*, *Salvar programa* e *Abrir programa*.

### Compilar programa

O caso de uso *Compilar programa* ocorre quando o usuário já escreveu o seu programa e deseja saber se existe algum erro de compilação. Como mencionado no item anterior, existem vários níveis de programação, portanto, a compilação depende de qual nível o programa foi escrito. Para compilar, o usuário deve apertar o botão *Compilar*, presente na tela. O processamento ocorre segundo as seguintes etapas:

- Análise do código-fonte na linguagem RoboEduc através do tradutor desenvolvido junto com o software RoboEduc;
- Verificação de erros;
- Reportar uma janela mostrando se existe erro ou não;
- Geração do código-fonte em linguagem C caso o nível escolhido seja diferente do nível 5;
- Verificação da existência de erros;
- Reportar uma janela mostrando se existe erro ou não.



### Enviar Comando

O caso de uso *Enviar comando* representa o envio de pacotes através do protocolo de comunicação usado no processador do robô (o RCX). Para que tais pacotes sejam enviados, torna-se necessário o clique no botão referente à ação correspondente.

### Enviar Programa

Este caso de uso diz respeito ao ato de enviar o programa já compilado e verificado para o RCX. Implicitamente, este caso de uso faz também a inicialização e verificação do protocolo de comunicação.

## 3.3 Implementação do RoboEduc

A implementação da ferramenta foi realizada no ambiente de programação C++. O Qt versão 4.3 foi a biblioteca utilizada para a implementação das interfaces com o usuário. O XML foi a linguagem de marcação utilizada para tornar o sistema dinâmico. A biblioteca BrickOS (SO e compilador), foi a API utilizada para programar os códigos-fontes dos robôs que recebem as informações enviadas pelo RoboEduc.

Na versão 1.0, existiam apenas as funcionalidades de *Controle* e comunicação entre o programa e o robô, porém essa comunicação era realizada por porta serial. Na versão 2.0, além do *Controle* foi implantada a funcionalidade *Ensinar protótipo* e a comunicação entre o robô e o programa passou a ser realizada pela porta USB. Para a versão 3.0, nossa proposta de trabalho, implementamos, além das funcionalidades anteriores, os níveis de programação descritos no *Caso de Uso*. Após a modelagem das novas funcionalidades, iniciamos a implementação das classes utilizando C++ e, em paralelo, as interfaces com o usuário em Qt.

Este software é bastante dinâmico. A partir da modelagem adotada, quando é necessário adicionar qualquer conteúdo ao sistema, esse conteúdo é armazenado em um arquivo XML. Portanto, não existe a necessidade de re-compilar o programa. A estrutura do arquivo XML funciona como um repositório de dados, sendo mostrada na Figura 3.5.

Um arquivo XML é lido na tela inicial de cada módulo *Aluno e Autoria*, uma vez que este contém as informações dos nomes, das descrições dos componentes, os endereços das imagens animadas dos modelos, das bases, dos atuadores e de todos os outros componentes, como também todas as funcionalidades cadastradas. Ao ler este arquivo XML, a classe escrita em Qt faz o uso do mecanismo *DOM*. Este mecanismo proporciona que as informações sejam extraídas do XML e armazenadas em uma árvore na memória. A partir desta árvore, as informações são armazenadas em um objeto da classe *QList*. Para organizar as informações em listas, foram criadas classes com os campos necessários e os tipos das listas foram definidos de acordo com o tipo de objeto necessário.

Todas as classes que contêm imagens tipo *gif* ou informações referentes ao conteúdo do software, foram criadas através da manipulação de arquivos XML. Essa característica torna o software um software de Autoria, facilmente extensível e dinâmico. O software contém uma linguagem de programação própria, denominada RoboEduc, cujos progra-

```

<?xml version="1.0" encoding="UTF-8"?>
<RoboEduc>
  <modelo>
    <nome>Corredor</nome>
    <imagem>figura/modelos/corredorPrototipo.gif</imagem>
    <descricao>essa e a descricao do corredor</descricao>
    <base>
      <gif>figura/modelos/corredorBase.gif</gif>
      <nome>Base do Corredor</nome>
      <descricao>essa eh a descricao da base</descricao>
    </base>
    <atuador>
      <gif>figura/modelos/naopossui.gif</gif>
      <nome>NaoPossui</nome>
      <descricao>Este modelo NaoPossui atuador</descricao>
    </atuador>
    <sensor>
      <gif>figura/modelos/naopossui.gif</gif>
      <nome>NaoPossui</nome>
      <descricao>Este modelo NaoPossui sensor</descricao>
    </sensor>
    <acao>
      <gif>figura/setaCima.gif</gif>
      <nome>Frente</nome>
      <descricao>O robo se movimenta para a frente</descricao>
      <composicao>
        <item>motor_a_dir(fwd)</item>
        <item>motor_c_dir(fwd)</item>
      </composicao>
    </acao>
    <sentidos>
      <gif>figura/setaCima.gif</gif>
      <nome>NaoPossui</nome>
      <descricao>Este robo NaoPossui sentidos</descricao>
      <composicao>
        <item>'sensorA luz'</item>
      </composicao>
    </sentidos>
  </modelo>

```

Figura 3.3: Estrutura do arquivo XML



mas em forma textual possuem terminação *.rob*. A necessidade da criação desta linguagem ocorreu pela complexidade encontrada nas linguagens de programação já existentes. Para a implementação desta linguagem, foi desenvolvida uma gramática própria da linguagem, descrita a seguir.

### 3.3.1 Gramática da Linguagem

As produções são as seguintes:

- $\langle \text{tarefa} \rangle := \text{tarefa } \langle \text{nome da tarefa} \rangle \text{ inicio } \langle \text{instrucao} \rangle^* \langle \text{fim} \rangle$
- $\langle \text{nome da tarefa} \rangle := \text{letra}(\text{letra}/\text{numero})^*$
- $\langle \text{controle de fluxo} \rangle := \langle \text{condicional} \rangle \parallel \langle \text{repeticao} \rangle$
- $\langle \text{condicional} \rangle := \text{se } \langle \text{condicao} \rangle \text{ então } \langle \text{instrucao} \rangle^* [\text{senão } \langle \text{instrucao} \rangle^*] \text{ fimse}$
- $\langle \text{condicao} \rangle := (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle) \parallel ((\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle) \langle \text{op logico} \rangle (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)) \parallel (\text{não } (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)) \parallel ((\text{não } (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)) \langle \text{op logico} \rangle (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)) \parallel ((\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle) \langle \text{op logico} \rangle (\text{não } (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle))) \parallel ((\text{não } (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)) \langle \text{op logico} \rangle (\text{não } (\langle \text{variavel} \rangle \langle \text{op} \rangle \langle \text{numero} \rangle)))$
- $\langle \text{variavel} \rangle := \text{letra}(\text{letra}/\text{numero})^*$
- $\langle \text{acao} \rangle := \text{executar } \langle \text{nome da tarefa} \rangle \parallel \langle \text{nome de acao} \rangle \langle \text{numero} \rangle \text{ segundos}$
- $\langle \text{repeticao} \rangle := \langle \text{repita para} \rangle \parallel \langle \text{enquanto} \rangle$
- $\langle \text{repita para} \rangle := \text{para } \langle \text{variavel} \rangle := \langle \text{numero} \rangle \text{ ate } \langle \text{numero} \rangle \text{ faça } \langle \text{instrucao} \rangle^* \text{ fimpara}$
- $\langle \text{enquanto} \rangle := \text{enquanto } \langle \text{condicao} \rangle \text{ faça } \langle \text{instrucao} \rangle^* \text{ fimEnquanto}$
- $\langle \text{op} \rangle := = \parallel > \parallel \geq \parallel < \parallel \leq$
- $\langle \text{op logico} \rangle := \text{e } \parallel \text{ou}$
- $\langle \text{letra} \rangle := [\text{A-Za-z}]$
- $\langle \text{instrucao} \rangle := \langle \text{controle de fluxo} \rangle \parallel \langle \text{acao} \rangle$
- $\langle \text{nome de acao} \rangle := \text{letra}(\text{letra}/\text{numero})^*$
- $\langle \text{numero} \rangle := 0 \parallel 1 \parallel 2 \parallel 3 \parallel 4 \parallel 5 \parallel 6 \parallel 7 \parallel 8 \parallel 9^*$

### 3.3.2 As Palavras Reservadas da Linguagem

As palavras reservadas usadas pela nossa linguagem são mostradas na Figura 3.4. Como podemos ver, são poucas palavras usadas pela linguagem, mostrando assim um alto nível de abstração. Apesar de ter um nível bem próximo da linguagem infantil, esta linguagem permite ensinar estruturas algorítmicas, tais como início de programa, fim de programa, controle de fluxo, comandos condicionais, chamadas de funções (elementares) e outras estruturas mais simples, como o envio de comandos simples.

### 3.3.3 O interpretador do RoboEduc

A partir da definição da gramática acima, a etapa seguinte foi a implementação do interpretador da linguagem RoboEduc para a linguagem C e BrickOS. As etapas da im-

Palavras Reservadas	
tarefa	repita
inicio	enquanto
fim	fimenquanto
se	para
entao	ate
fimse	faca
senao	fimpara
fimsenao	e
executar	ou
nao	

Figura 3.4: As palavras reservadas da linguagem RoboEduc

plementação foram as seguintes:

- Implementação do Analisador Léxico.
- Implementação do Analisador Sintático.

No analisador léxico, a entrada é um arquivo *.rob* na linguagem RoboEduc onde separa o arquivo em tokens. No analisador sintático os tokens são recebidos do analisador léxico e é analisado a estrutura do arquivo a fim de verificar se esta de acordo com as regras da gramática definida. Uma vez analisado sintaticamente, é iniciado o processo de interpretação da linguagem RoboEduc para linguagem C e BrickOS. A saída do analisador sintático é um arquivo escrito em C e BrickOS. Após esta etapa faz-se uma chamada de sistema para o compilador BrickOS, que compila o arquivo C. A saída do compilador BrickOS é um arquivo *.lx* que é enviado ao robô. Todas as etapas de tradução reportam mensagens de erros caso eles ocorram.

### 3.3.4 Diagramas UML: Classe

O diagrama de classes demonstra a estrutura estática das classes de um sistema onde estas representam os objetos que são gerenciadas pela aplicação modelada. Classes podem se relacionar com outras através de diversas maneiras: associação (conectadas entre si), dependência (uma classe depende ou usa outra classe), especialização (uma classe é uma especialização de outra classe), ou em pacotes (classes agrupadas por características similares). Todos estes relacionamentos são mostrados no diagrama de classes juntamente

com as suas estruturas internas, que são os atributos e operações. O diagrama de classes é considerado estático já que a estrutura descrita é sempre válida em qualquer ponto do ciclo de vida do sistema. Um sistema normalmente possui alguns diagramas de classes, já que não são todas as classes que estão inseridas em um único diagrama e certa classe pode participar de vários diagramas de classes. O diagrama de classes mostra uma representação conceitual das informações, que abrangem uma coleção de elementos declarativos válidos nos diversos estágios do sistema. A implementação do sistema RoboEduc seguiu o seguinte diagrama de classes.

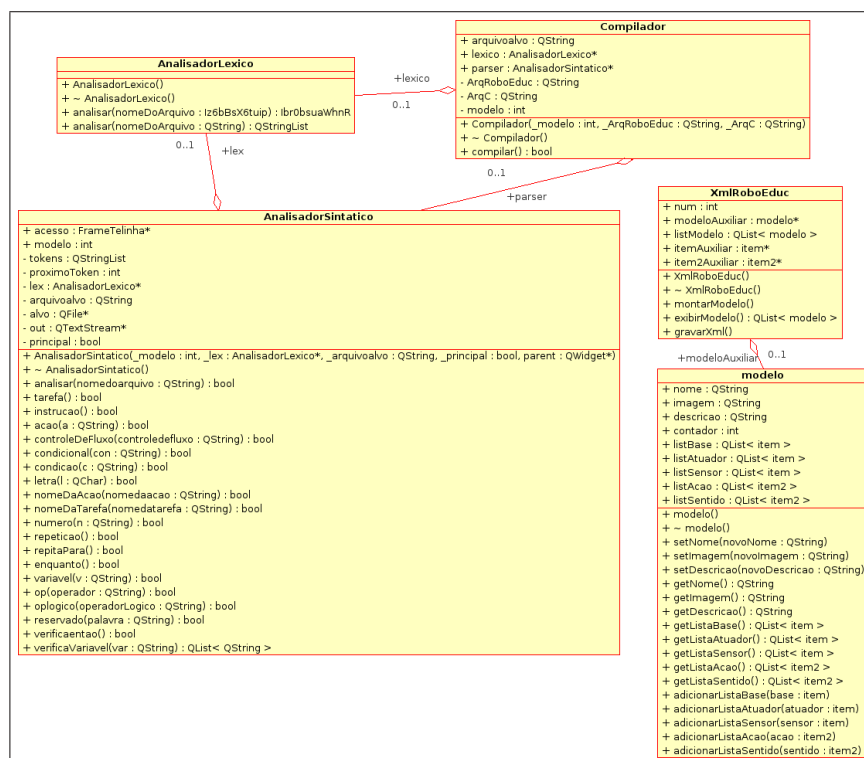


Figura 3.5: Diagrama das principais classes do RoboEduc

A Figura 3.5 ilustra o diagrama de classes construído para o RoboEduc. Este diagrama representa uma visão geral dos dados através das classes principais. Cada janela do sistema foi implementada como uma classe. Foge do escopo deste texto a descrição detalhada de cada classe, sendo esta tarefa executada quando da modelagem UML do nosso programa. Assim, colocamos basicamente as classes de manipulação do XML e do interpretador, que são as mais importantes. Outras classes como de interface e interação com o usuário fazem parte do Qt 4.3. Olhando a Figura 3.5, podemos perceber que as classes sintetizam as funcionalidades descritas acima, nos casos de uso. Nos experimentos, serão mostradas funcionalidades práticas, com a execução dos métodos definidos nessas classes.

---

## Capítulo 4

# Experimentos e Resultados

---

### 4.1 Experimentos

Os experimentos descritos a seguir se referem as versões anteriores do RoboEduc. Na versão 2.0 do RoboEduc, foi implantado o Projeto de Inclusão Digital com Robôs. Este projeto foi desenvolvido em uma instituição pública do estado do Rio Grande do Norte, a Escola Municipal Professor Ascendino de Almeida, localizada em um bairro periférico do município de Natal. Como toda escola pública, esta instituição não diverge das demais, apresenta uma realidade escolar deficitária e alunos oriundos de famílias com situações econômicas e sociais precárias.

É nesse contexto que o Projeto de Inclusão Digital com Robôs está inserido. Para facilitar o aprendizado, as crianças foram divididas em grupos e cada grupo possuiu um monitor para supervisionar as atividades. Ao todo são 24 crianças que participavam do projeto: 12 meninos e 12 meninas. Foram escolhidos os alunos da terceira e quarta séries do Ensino Fundamental.

As aulas de robótica foram ministradas por meio de oficinas pedagógicas e os objetivos do Projeto estão sendo alcançados paulatinamente, mediante o trabalho das monitoras, inclusive fui uma das monitoras, instruídas para proporcionar o conhecimento de maneira acessível às crianças. As oficinas eram realizadas através de um planejamento detalhado de nossas ações nas oficinas pedagógicas e de um sistema avaliativo contínuo dentro e fora da sala de aula. Este projeto visa apresentar uma proposta para utilizar robôs como ferramenta de apoio ao aprendizado colaborativo de pequenos grupos de pessoas no mesmo lugar geográfico e com interações síncronas.

O planejamento das oficinas eram realizadas em reuniões semanais, onde eram discutidos os erros e acertos, a postura dos alunos e a compreensão dos conteúdos de robótica e informática pelos mesmos. Era avaliada, também, a postura das monitoras diante dos problemas enfrentados. Essas reuniões serviam como referências para as próximas oficinas, assim, caso necessário, era realizada mudança de estratégias e de postura diante dos alunos.

No ano de 2006 foram realizadas 10 (dez) oficinas. Os alunos não eram somente motivados a montar e manipular o robô através do programa RoboEduc, mais também a participar de atividades em que envolvem novos conceitos, como, noção espacial, velocidade, noção de números, entre outros.

Inicialmente os alunos receberam noções iniciais de robótica e de informática. Depois foi proposto, um conjunto de tarefas (banco de tarefas), incluídas no RoboEduc, similares a jogos educativos, mas, com o diferencial de utilizar robótica como auxílio ao aprendizado, que deveriam ser executadas de forma colaborativa.

As atividades eram colocadas como situações problemas que não conduzissem os alunos somente a manipular os legos, mas também, torna-os construtor do seu robô, através da colaboração, construção e reconstrução em parceria com os colegas e monitores. Desta forma, o trabalho colaborativo foi o centro do desenvolvimento das atividades. Por isso, todas as atividades foram desenvolvidas em grupo e o papel de cada monitor era estimular os alunos a refletir sobre as decisões tomadas, sobre a importância da colaboração, promovendo o diálogo e o respeito a diferentes opiniões. Dentre estas tarefas podemos citar: Jogo da forca, jogo da velha, formar diagramas, quebra-cabeça, construção de mapas, desenhar objetos etc.

Os alunos foram divididos em grupos de 6 componentes, de séries diferentes e sempre acompanhados por monitores. Cada grupo possuía um kit Lego sobre sua responsabilidade. As atividades foram planejadas de modo que favorecessem o espírito crítico, a criatividade e a capacidade de tomadas de decisões. Os conteúdos das oficinas eram estabelecidos antes do início das mesmas, no entanto, eram bastante flexíveis, podendo ocorrer alterações no cronograma de acordo com o grupo.

As oficinas funcionavam da seguinte maneira: escolhia-se uma tarefa para ser executada através do ambiente desenvolvido (RoboEduc), o instrutor passava a ensinar e a auxiliar o primeiro usuário que deveria, dependendo do objetivo, controlar e repetir um, determinado comportamento ou programar o robô. Um aluno, ao concluir sua parte da tarefa, ensinava ao próximo a manusear a ferramenta desenvolvida e a interagir com o sistema, levando a uma transmissão de conhecimento entre as crianças. Ou seja, as atividades eram feitas pelas crianças através dos protótipos, proporcionando uma interatividade constante entre essas duas entidades até a conclusão do objetivo proposto. Cada jogo proposto possuía um objetivo definido, envolvia algum assunto de interesse acadêmico. Além disso, todos os assuntos requeriam discussões do grupo a respeito da forma como finalizá-la. Vale salientar que, antes de utilizar o sistema para completar a tarefa, os alunos (em grupos) constroíam, interativamente, um robô utilizando o kit Lego com o auxílio do instrutor.

A cada dia de experimentos era apresentado um modelo de robô já definido e, então, os alunos montavam. Quando cada grupo terminava suas atividades, as crianças tinham a liberdade para fazer o que quisessem com os robôs é como se elas “brincassem” com o robô. Assim verificou-se que os robôs eram utilizados como ferramentas de aprendizado bastante atrativas, tanto por serem encarados pelas crianças como brinquedos - elas pensavam estar simplesmente “brincando”, e não estudando. Logo, a aprendizagem com recreação, torna o entendimento do conteúdo mais eficiente por ser realizada de forma colaborativa, uma vez que podemos verificar que a maioria das atividades lúdicas são realizadas em grupo. As oficinas podem ser vista como mostrada nas Figura 4.1



Figura 4.1: Oficinas de Robótica

## 4.2 Resultados

O principal objetivo deste projeto é minimizar o problema da inclusão digital no RN, assim complementando projetos já existentes, começamos a pensar numa maneira de trabalhar com as mais digitalmente excluídas: crianças que moram na periferia e que são alunos do Ensino Fundamental.

Embora nosso projeto ainda esteja em andamento, é possível constatar várias mudanças no comportamento das crianças que participam desse projeto. As mudanças se estendem desde as oficinas de robótica e o desempenho na sala de aula até o comportamento em suas casas. Os experimentos estão utilizando o programa RoboEduc explicado no Capítulo 3 e a metodologia de ensino colaborativa.

Mediante as possibilidades apresentadas, nosso sistema avaliativo foi elaborado com a finalidade de avaliar os alunos participantes das oficinas de robótica pedagógica de forma contínua e com dados qualitativos. Tomamos por base que cada aluno possui interesses e ritmos de aprendizado distintos e que a robótica inseri-se nesse universo como forma de motivar esses alunos para aprendizado, uma vez que ela contempla o desenvolvimento pleno do aluno, pois propicia uma atividade dinâmica, permitindo a sua construção cultural e, enquanto cidadão, torna-o autônomo, independente e responsável [ZILLI 2004]. As avaliações foram efetuadas através de entrevistas com as professoras dos alunos participantes das oficinas, com a Diretora da instituição onde o Projeto foi efetivado e com as monitoras das oficinas de robótica educacional, resultando no conhecimento dos problemas do cotidiano de sala de aula e das oficinas como atividade extracurricular da escola.

A partir dos problemas conhecidos e visando, saná-los, bem como, desvendar os avanços ocasionados após a inserção da robótica na escola, foram realizadas considerações sobre os aspectos comportamentais e de conteúdo. Por meio dos dados colhidos das entrevistas, percebeu-se que as professoras dos alunos participantes das oficinas e a diretora da instituição estavam satisfeitas com a implantação do Projeto na escola. Os benefícios que o projeto está proporcionando foram imensos, inclusive, as professoras e diretora afirmaram que o Projeto possibilitou a essas crianças o contato com a tecnologia, algo até então desconhecidos para eles. A diretora da instituição de ensino, resaltou que este



Projeto é de fundamental importância, principalmente por que se trata de alunos carentes no sentido amplo da palavra, os quais não têm nem acesso aos meios tecnológicos de primeira necessidade, como televisão, telefone, geladeira, dentre outros. Ainda de acordo com a diretora, esse Projeto veio a desenvolver até a questão cognitiva dos alunos participantes. Por meio de entrevista com as professoras, elas revelaram que após a entrada da robótica na vida desses alunos, eles se tornaram mais atentos, o raciocínio lógico apresentaram melhoras e segundo uma das professoras entrevistadas, os alunos participantes tornaram-se mais responsáveis e comprometidos com as atividades em sala de aula.

Continuando com as entrevistas, foi declarado que as crianças participantes do Projeto Inclusão Digital com Robôs eram alunos com dificuldades na aprendizagem, com baixa concentração em sala de aula e alguns com repetências seguidas. A diretora expôs que essas crianças foram escolhidas criteriosamente, pois a comunidade escolar apostou na robótica como atividade motivadora para a aprendizagem desses alunos e, a mesma afirmou que, ocorreu resultados significativos. As professoras e a diretora da escola comentaram sobre a imagem que elas possuíam acerca do Projeto, ambas foram unânimes em dizer que houve um encantamento com o Projeto, principalmente depois que o conheceram e puderam constatar que as crianças iriam poder executar o que estavam aprendendo, que haveria ligação entre teoria e prática, assim, o conhecimento não iria ser transmitido de forma abstrata e subjetiva, mas sim, concretamente e de acordo com realidade daqueles alunos.

Por fim, foi ressaltado as expectativas futuras em relação ao Projeto de Inclusão Digital com Robôs. Não houve dúvidas em relação ao interesse da continuidade do Projeto na escola, correspondendo que a robótica realmente se inseriu como um recurso eficiente e facilitador no processo de ensino-aprendizado desses alunos. Essa avaliação nos mostrou a visão de pessoas externas ao Projeto, no caso, professoras e diretora da instituição de ensino onde o mesmo se desenvolveu. Esse avanço foi muito significativo para nós, pois obtivemos assim, o resultado da eficácia desta ferramenta num ambiente escolar através das pessoas que conhecem seus alunos, suas dificuldades e sabem identificar melhoria de desempenho dos alunos.

#### **Opinião das professoras da instituição de ensino acerca do Projeto de Inclusão Digital com Robôs.**

*“Essa foi uma idéia fantástica, fantástica mesmo. É um mundo novo para os nossos alunos. Eles que são tão limitados, eles ficam tão aquém de toda tecnologia que estar por aí. Fantástica a idéia.”*(PROFESSORA II)

*“Eu e com certeza toda a escola queremos a ampliação para que mais alunos pudessem participar desse projeto.”*(PROFESSORA I).

*“A expectativa que eu tenho é que não fique só por aqui, que ele, o projeto, continue por muitos e muitos anos.”*(PROFESSORA II).

#### **Opinião da diretora da instituição acerca do Projeto de Inclusão Digital com Robôs:**

*“Uma oportunidade dessas para nós foi maravilhosa e percebemos realmente que esta fazendo com que o professor e até a própria escola acorde, por que nós educadores*

*também somos analfabetos digitais e sem contar com a questão da robótica que já é um outro leque de conhecimentos, fala ainda: inclusive é uma forma até de estímulo aos próprios professores que de certa forma também estão afastado dessa inclusão digital por uma série de fatores.” (DIRETORA).*

*“O que eu sinto como diretora da escola, é que realmente precisa que o conjunto de professores se integre mais ao projeto, porque todos sabem que essas crianças estão vindo para as oficinas, mas ainda há um distanciamento de um modo geral entre o grupo de professores, a própria direção da escola e o projeto.” (DIRETORA).*

*[...] realmente falhamos, mas vamos tentar concertar, conversar com os professores e saber quem tem interesse voluntário de vir um dia à noite aqui na escola ou um outro horário que for conveniente para a equipe, para que eles se integrem mais com o projeto.(DIRETORA).*

*[...] a principio eu achei que fosse feito diretamente com os professores. Os professores iriam passar isso para as crianças.(DIRETORA).*

*“O que me deixou feliz, foi o fato de que era diretamente com as crianças e outra coisa, que as crianças iriam ter a oportunidade de executar o que elas iriam ver, não era uma coisa apenas teórica e subjetiva. Então realmente quando eles apresentaram, mostraram o objetivo do projeto, foi que eu percebi que a dimensão era maior do que a gente imaginava.”(DIRETORA).*

*“Hoje em função desse projeto, os meninos já colocam a cabeça na sala da direção e dizem: “Diretora quando é que a gente vai ter um computador para mexer?” Às vezes alguém esta digitando alguma coisa e eles pedem para olhar. Nós arranjamos uns teclados velhos e foi uma disputa grande, porque todas as salas queriam ter só o teclado! Então, nós não imaginávamos que estes alunos que estão vindo fossem ser multiplicadores e eles estão sendo sem nenhuma orientação. Eles chegam à sala de aula e comentam, conversam com os outros meninos sobre as oficinas. Tem meninos aqui, inclusive, que deixam de lanchar para pegar o dinheiro e ir a uma lan house para poder ter acesso ao computador. Para nós foi uma surpresa que o Projeto chegasse a estimular eles nesse sentido.”(DIRETORA).*

*“Eu espero que esse projeto dê pelo menos uma luz de capacitação técnica para os professores. Eu vejo que o professor precisa entrar nessa proposta para aprender a desenvolver a inclusão digital no sentido de ajudar no trabalho de sala de aula. Por que nós não podemos mais permanecer com giz e quadro, por que percebemos que o aluno não esta mais respondendo a essa limitação, embora eles sejam alunos carentes, há todo um universo de informações disponíveis para eles no contexto da vida cotidiana.”(DIRETORA).*

*“Apesar de todos os problemas que estas crianças têm, elas também possuem um conjunto de conhecimentos grande e o quadro e o giz não estão mais conta disso. Então a nossa esperança nesse tipo de projeto é muito grandiosa, por que sempre falamos que vamos partir da realidade das crianças, mas na verdade nós na partimos da realidade delas. A realidade delas é muito ampla e a gente não estar trazendo esta realidade para a sala de aula. O que é olhar um quadro todo escrito em giz para uma criança que fica o tempo todo em frente a uma televisão, para uma criança que vive correndo no meio da rua e brincando? Então eles são dispersos dentro de uma sala de aula, mas*



*são extremamente concentrados para esse tipo de atividade, então certamente eles já não estão mais se identificando com essa metodologia. A nossa esperança é que esse tipo de projeto faça a gente romper com a utilização permanente do quadro e do giz e encontrar caminhos para enriquecer todas as atividades de sala de aula.”(DIRETORA).*

### 4.3 Usando o RoboEduc versão 3.0

Neste experimento, ilustramos o passo a passo da utilização da nova versão do software RoboEduc. A Figura 4.2 mostra a tela inicial do RoboEduc, de onde iniciaremos um exemplo de construção de protótipo.



Figura 4.2: Tela Inicial do RoboEduc

A partir da tela inicial, é acessado o módulo autoria, mostrado na Figura 4.3, de onde é possível criar novos modelos, bases, atuadores, sensores, ações em alto nível (andar para a frente, trás, direita, esquerda etc). A Figura 4.4 ilustra o uso do RoboEduc na operação de montagem do protótipo. No caso, um protótipo com rodas, dois motores, que executa cinco funcionalidades foi escolhido. Neste caso, estas funcionalidades são: andar para frente, andar para trás, girar para a direita, girar para a esquerda e parar.

Após escolhido um modelo (protótipo), são apresentadas ao usuário as opções de construção para ele, cadastrado pelo professor previamente, e então, o usuário (aluno, agora) escolhe os componentes existentes no protótipo montado, como mostrado na Figura 4.5. Após esta etapa é apresentada ao usuário uma tela com a descrição do robô, a tarefa que ele deve executar e uma imagem tipo *gif animado* do protótipo escolhido. A próxima etapa será enviar o programa de escuta ao robô, escolhida pelo botão *Programa*. Então, é estabelecida a comunicação entre o robô e o computador, conforme mostrado na Figura 4.6.

A seguir, pela escolha da funcionalidade Controlar protótipo, serão exibidas as ações que o protótipo pode executar, conforme mostrado na Figura 4.7. Ao pressionar qualquer um dos botões correspondentes às ações, o comando correspondente será enviado ao robô.



Figura 4.3: Tela para criar modelos, bases, atuadores, sensores, ações, sentidos



Figura 4.4: Tela para Criar Protótipos



Figura 4.5: Selecionar Protótipos Cadastrados



Figura 4.6: Selecionar o que se deseja fazer com o protótipo

Isto é, um simples clique no botão proporciona ao robô andar para frente, girar para esquerda, entre outras. Notamos aqui, novamente, que cada botão representa uma ação que foi previamente cadastrada para o protótipo selecionado.



Figura 4.7: Tela de controle do protótipo

A tela mostrada na Figura 4.8 aborda, além das funcionalidades descritas no caso de uso Controlar Protótipo, descrito no Capítulo 3, a opção de armazenar os comandos executados em um arquivo e torna-lo apto a ser utilizado posteriormente. Uma outra funcionalidade é abrir um arquivo existente e executá-lo. Após a escolha de uma série de comandos, os mesmos podem ser enviados ao robô, conforme mostrado na Figura 4.8, ou então, os mesmos podem ser gravados, usando armazenar, acima.

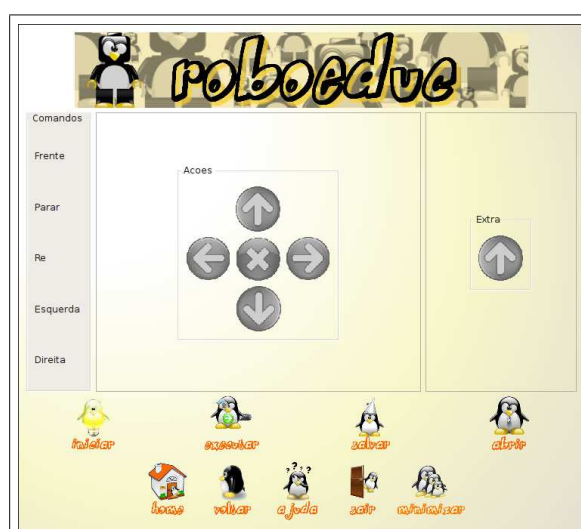


Figura 4.8: Tela para ensinar o protótipo

A Figura 4.9 mostra a tela com os vários níveis de programação. Todos os níveis já se encontram implementados. Ao escolher um dos níveis, a interface de programação correspondente é mostrada. As Figuras 4.10, 4.11, 4.12, 4.13 e 4.14 mostram as respectivas interfaces, para cada um dos níveis (1 a 5).



Figura 4.9: Tela dos Níveis de Programação



Figura 4.10: Tela do Nível 1 de Programação



Figura 4.11: Tela do Nível 2 de Programação



Figura 4.12: Tela do Nível 3 de Programação



Figura 4.13: Tela do Nível 4 de Programação



Figura 4.14: Tela do Nível 5 de Programação



---

## Capítulo 5

### Conclusões

---

No presente trabalho, propomos uma nova versão para o software educacional RoboEduc, visando principalmente levar conceitos de computação, construção de robôs e programação para crianças na faixa etária de 06 a 10 anos. Visando diminuir a complexidade de programação desses softwares, principalmente para poder torná-lo compreensível por parte das crianças, bem como de seus professores (leigos), desenvolvemos nesta nova versão uma série de melhorias, incluindo uma nova linguagem de programação de robôs, a ser usada pelos usuários do RoboEduc. Esta linguagem consta de interpretador, compilador, visando atender vários níveis de programação, desde comandos mais elementares, entrados via teclado ou interface visual, até o controle de estruturas de fluxos. Na parte experimental, o software foi validado com uma sequência de telas, apresentando as várias funcionalidades do mesmo.

Entre as nossas principais contribuições no presente trabalho, destacamos a participação em experimentos (oficinas) de interação entre os robôs e as crianças na Escola Municipal Ascendino de Almeida e a implementação do software para controle e programação de plataformas robóticas, facilmente manipulável por crianças do primeiro ao quinto ano escolar. Convém ressaltar que participamos desde a primeira versão do software, hoje totalmente modificada em relação à sua versão original. Colaboramos ainda no desenvolvimento da linguagem de programação completa para programação de robôs, incluindo interpretador e compilador, visando efetivar a programação dos robôs pelas crianças. Colaboramos também na confecção de artigos que foram apresentados em conferências nacionais e internacionais do grupo [Silva et al. 2008, Barrios-Aranibar et al. 2006].

Convém ressaltar que o presente trabalho faz parte de um projeto maior, que se trata inserção da Robótica Educacional em escolas da rede pública, desenvolvido no Laboratório Natalnet, onde estão envolvidos vários outros alunos. Assim, os nossos desenvolvimentos serão essenciais para a continuidade deste projeto, dentro do referido Laboratório.

O protótipo inicial do software RoboEduc foi utilizado na alfabetização digital. Em geral, o público alvo do RoboEduc são crianças do primeiro ao quinto ano do ensino público, que tiveram pouco contato com esse tipo de tecnologia. Os próximos passos são a realização de experimentos práticos visando efetivar nossa metodologia em outras escolas. A título ilustrativo, o Colégio Contemporâneo já foi contatado nesse sentido e os seus gestores mostraram-se interessados em nosso trabalho, sendo uma de nossas futuras escolas parceiras. Implementações futuras referem-se ao manual on-line do software, manuais do aluno e do professor, bem como melhorias no sub-sistema tutor.



---

## Referências Bibliográficas

---

- Asimov, ISAAC (n.d.), Isaac asimov biografia, [http://www.din.uem.br/ia/a\\_correl/classicos/Pesquisadores-Asimov.htm](http://www.din.uem.br/ia/a_correl/classicos/Pesquisadores-Asimov.htm).
- Bagio, Rodrigo (2003), 'Mapa da exclusão digital'. Last Access in 20/04/2008.  
**URL:** [http://www2.fgv.br/ibre/cps/mapa\\_exclusao/apresentacao/apresentacao.htm](http://www2.fgv.br/ibre/cps/mapa_exclusao/apresentacao/apresentacao.htm)
- Barrios-Aranibar, Dennis, Viviane Gurgel, Aquiles Burlamaqui, Luiz M. G. Gonçalves, Marcela Santos, Gianna R. Araújo, Válber C. Roza & Rafaella A. Nascimento (2006), Technological inclusion using robots, *em* 'Anais do II ENRI - Encontro Nacional de Robótica Inteligente', XXVI Congresso da Sociedade Brasileira de Computação - SBC2006, Campo Grande, MS, Brazil.
- Bri (2008), 'An open source embedded operating system and provides a 'c' and 'c++' programming environment for the lego mindstorms robotics kits', <http://brickos.sourceforge.net>. Last Access in 30/04/2008.
- Cyb (2008), 'Cyberbox', <http://www.cyberbox.com.br/cb.php>. Last Access in 15/05/2008.
- da Silva Filho, Antonio Mendes (2003), 'Os três pilares da inclusão digital', *Revista Espaço Acadêmico* **3**(24).
- dos Santos, Marcela Gonçalves (2006), 'Roboeduc: Um software para ensino da robótica para crianças digitalmente excluídas utilizando protótipos lego', *Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) - Universidade Federal do Rio Grande do Norte*. (24).
- D.W.Carraher (1990), 'O que esperamos do software educacional?', *In Revista de Educacao e Informatica*. **3**(24).
- Edu (2008), 'Educabrazil', <http://www.educabrazil.com.br>. Last Access in 17/05/2008.
- Fitch, D. (2002), Digital inclusion, social exclusion and retailing: an analysis of data from the 1999 scottish household survey, *em* 'Technology and Society, 2002. (ISTAS'02). 2002 International Symposium on', pp. 309– 313.
- Gonçalves, Luiz M. G., Douglas Machado Tavares & Viviane André Antunes (2005), Em evidência o potencial e limitações dos compiladores nqc e brickos e seus respectivos sistemas operacionais.

- Inf (2008a), 'O mundo da informática na escola', <http://www.informareducacional.com.br>. Last Access in 10/06/2008.
- Inf (2008b), 'Tux4kids | high quality educational software alternatives', <http://www.tux4kids.com>. Last Access in 10/06/2008.
- LEG (2008), 'Lego mindstorms', <http://mindstorms.lego.com>. Last Access in 20/05/2008.
- Lej (2008), 'Java for lego mindstorms', <http://lejos.sourceforge.net>. Last Access in 10/03/2008.
- LNP (2008), 'Legos network protocol', <http://legos.sourceforge.net/HOWTO/x405.html>. Last Access in 30/04/2008.
- Meer, J. (2003), 'Getting on the net: the struggle for digital inclusion of the navajo', *Technology and Society Magazine, IEEE* **22**(1), 53–58.
- NQC (2008), 'Nqc', <http://bricxcc.sourceforge.net/nqc>. Last Access in 10/05/2008.
- ONU (2008), 'Onu', <http://www.onu-brasil.org.br>. Last Access in 10/03/2008.
- PNC (2008), 'Alfa kit's para robótica', [http://www.pnca.com.br/index.php?option=com\\_frontpage&Itemid=1](http://www.pnca.com.br/index.php?option=com_frontpage&Itemid=1). Last Access in 17/05/2008.
- Porcaro, Rosa Maria (2006), 'Tecnologias de comunicação e informação: políticas e estratégias de inclusão digital no brasil'. Last Access in 02/05/2008.  
**URL:** <http://www.ipea.gov.br/>
- Qt (2008), 'Qt trolltech', <http://doc.trolltech.com/4.3/index.html>. Last Access in 30/05/2008.
- RCT (2008), 'Software educativo', <http://www.rctsoft.com.br>. Last Access in 10/06/2008.
- Re- (2008), 'Tecnologia educacional', <http://www.educasoft.com.br/Telas/recriar.htm>. Last Access in 10/06/2008.
- rob (2008), 'robokit', <http://www.imply.com.br/not2007/prtnot-robokit.htm>. Last Access in 15/05/2008.
- Rondelli, Elizabeth (2003), 'Quatro passos para a inclusão digital'. Last Access in 30/04/2008.  
**URL:** <http://www.comunicacao.pro.br/setepontos/>
- Selwyn, Neil (2004), 'Reconsidering political and popular understandings of the digital divide', *New Media Society* **6**, 341–362.
- Silva, Alzira F., Akynara Silva, Luis M. G. Goncalves, Guerreiro, Renata. B. Pitta, Ana Maria G. & D. Barrios-Aranibar (2008), Utilização da teoria de vygotsky em robótica educativa, em 'IX Congresso Iberoamericano De Informatica Educativa RI-BIE', IX Congresso Iberoamericano De Informatica Educativa RIBIE 2008, 2008., Caracas.

- Sof (2008), 'Software educativo', <http://www.softmarket.com.br>. Last Access in 10/06/2008.
- STEFFEN, Heloisa Helena. (2002), Robótica pedagógica na educação: Um recurso de comunicação, regulação e cognição, *em* 'Dissertação de Mestrado em Ciências da Comunicação. Escola de Comunicação e Artes, Universidade de São Paulo, SP, 2002.', [http://www.bibvirt.futuro.usp.br/textos/teses/helo\\_robotica.pdf](http://www.bibvirt.futuro.usp.br/textos/teses/helo_robotica.pdf).
- Trindade, Maria Luiza (2007), 'Produção de material e metodologias para o projeto de inclusão digital usando robôs', *Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) - Universidade Federal do Rio Grande do Norte*. (24).
- Willis, Suzanne & Bruce Tranter (2006), 'Beyond the 'digital divide': Internet diffusion and inequality in australia', *Journal of Sociology* **42**(1), 43–59.
- Wilson, Kenneth R., Jennifer S. Wallin & Christa Reiser (2003), 'Social stratification and the digital divide', *Social Science Computer Review* **21**(2), 133–143.
- ZILLI, Silvana do Rocio (2004), A robotica educacional no ensino fundamental: Perspectivas e praticas, *em* 'Dissertacao (Mestrado em Engenharia de Producao -Programa de Pos-Graduacao em Engenharia de Producao, Universidade Federal de Santa Catarina)'