

A ROBÓTICA COMO INSTRUMENTO DE ENSINO NAS ESCOLAS PÚBLICAS

Autor (1): José Torres Coura Neto; Co-autor (2): Fernando Costa Fernandes Gomes; Orientador (3): Euler Cássio Tavares de Macêdo

(1) *Universidade Federal da Paraíba, jose.neto@cear.ufpb.br*

(2) *Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, fernando.gomes@ifpb.edu.br*

(3) *Universidade Federal da Paraíba, euler@cear.ufpb.br*

RESUMO: o presente projeto tem como objetivo o desenvolvimento de um kit educacional de robótica utilizando as plataformas Arduino e Android para a realização de atividades de ensino na disciplina de física do ensino médio em escolas públicas. Para tanto, foi montado, a baixo custo, um robô denominado Ziroba que se assemelha a um carro de resgate de vítimas de desastre e que utilizou a placa de desenvolvimento Arduino, sendo possível seu controle e comunicação via bluetooth utilizando um smartphone contendo o aplicativo Controle Arduino, desenvolvido na plataforma App Inventor MIT. Após o desenvolvimento dos materiais foi possível a realização do I Desafio de Robótica no Instituto Federal de Educação, Ciência e Tecnologia – IFPB Campus Picuí com os alunos do 3º ano do Ensino Médio Integrado com o Curso Técnico de Geologia. Essa atividade proporcionou a discussão de temas relacionados com a disciplina de uma maneira lúdica e divertida, facilitando a compreensão do assunto e uma aprendizagem interdisciplinar.

Palavras-chave: Robótica, Física, Arduino.

1. INTRODUÇÃO

A educação básica no país, segundo o Censo Escolar 2013, possui cerca de 50 milhões de alunos matriculados, sendo que, na sua grande maioria, 82,8%, em escolas públicas. Dentre os quais, apenas 0,6% na rede federal. Ainda segundo o Censo, quando comparados os números de alunos matriculados no ensino médio e o da população na faixa-etária entre 15 e 17 anos, se observa que cerca de 2 milhões de possíveis alunos estão fora da sala de aula. Para mudar esse quadro, é necessária a implantação de políticas que estimulem o jovem concluinte do ensino fundamental a progredir nos seus estudos. Uma dessas políticas é a diversificação curricular, na qual atividades como a robótica educacional podem ser desenvolvidas.

A robótica educacional é uma atividade que desperta o interesse dos envolvidos com o intuito de desenvolver estruturas de hardware e software a fim de solucionar alguma situação-problema, Miranda *et al.* Segundo Schons et al., a robótica pedagógica “constitui nova ferramenta que se encontra à disposição do professor, por meio da qual é possível demonstrar na prática muitos dos conceitos teóricos, às vezes de difícil compreensão, motivando tanto o professor como principalmente o aluno”. De acordo com Zilli, a robótica educacional pode desenvolver as seguintes competências: raciocínio lógico; formulação e teste de hipóteses; relações interpessoais; investigação e compreensão; representação e comunicação; resolução de problemas por meio de

erros e acertos; aplicação das teorias formuladas a atividades concretas; criatividade; e capacidade crítica.

Os kits de robótica educacional disponíveis no mercado, apesar de serem muito eficientes quanto à facilidade de utilização por parte de alunos e professores, possuem preço elevado para aquisição, especialmente por conta de serem, em sua maioria, produtos de importação. Com o intuito de suprir essa dificuldade, o presente projeto propõe a utilização de plataformas de hardware e software livres e de baixo custo, como a placa de desenvolvimento Arduino e a plataforma Android, a fim de serem utilizados na realização de atividades de robótica nas escolas.

Esse trabalho está organizado da seguinte forma: na metodologia são abordadas as ferramentas de hardware e software utilizadas no projeto e o Desafio de Robótica realizado para os alunos; nos resultados são apresentadas as impressões dos alunos acerca da atividade desenvolvida; Na discussão, são analisados os resultados alcançados; E, por fim, a conclusão expõe as considerações finais e a sugestão de trabalhos futuros;

2. METODOLOGIA

Nessa etapa, serão descritas as ferramentas de hardware e software utilizadas no trabalho para a confecção do kit de robótica.

2.1. Construção do robô Ziroba

O robô Ziroba é uma versão do carro desenvolvido pelo Professor Isaac Maia da UFPB, que simula um carro real para resgate de vítimas de desastres, conforme a Figura 1.



Figura 1. Robô Ziroba 2.0.

O robô é construído utilizando materiais de baixo custo. Seu chassi é composto de caixas de luz, material utilizado na construção civil. E seu hardware é baseado em Arduino. Abaixo, segue uma tabela com os materiais utilizados e seu preço de mercado.

Tabela 1. Materiais e Custos do Ziropa.

Material	Preço
2 Caixas de luz 4x4	R\$ 8,00
1 Caixa redonda de luz	R\$ 4,00
1 Kit com 4 motores e caixas de redução	R\$ 86,00
1 Garra Robótica com 2 servo motores	R\$ 220,00
1 Arduino Uno R3	R\$ 44,00
1 Shield Motor	R\$ 28,00
1 Módulo Bluetooth	R\$ 28,00
1 Bateria Chumbo 6V	R\$ 50,00
1 Chave On/Off	R\$ 2,00
TOTAL	R\$ 470,00

Arduino é uma plataforma de desenvolvimento open-hardware que possui seu próprio ambiente de desenvolvimento, IDE, baseado na linguagem de programação C. O que torna a sua utilização bastante intuitiva para iniciantes. O software é compatível com os principais sistemas operacionais do mercado: Mac OS X, Windows e Linux. Suas especificações são: Microcontrolador: ATmega328 (Atmel); Tensão de Operação: 5V; Tensão de Entrada: 7-12V; Portas Digitais: 14 (6 podem ser usadas como PWM); Portas Analógicas: 6; Corrente Pinos I/O: 40mA; Corrente Pinos 3,3V: 50mA; Memória Flash: 32KB (0,5KB usado no bootloader); SRAM: 2KB; EEPROM: 1KB; Velocidade do Clock: 16MHz; A placa do Arduino pode ser observada na Figura 2.



Figura 2. Placa Arduino Uno R3.

Para a conexão dos motores com o Arduino, foi utilizado o Motor Shield L293D Driver Ponte H para Arduino. Tal circuito elétrico é utilizado para transferir energia de uma fonte de tensão para

uma carga em qualquer direção, como o motor DC utilizado no Ziroba. Esse Motor Shield é baseado no Chip L293D. O chip L293D possui internamente 2 Ponte H e suporta uma corrente de saída de 600mA por canal, sendo possível controlar até 2 motores com 600mA cada, visto que neste Shield temos 2 chips. O mesmo possui as seguintes especificações: Chip: 293D; Pode controlar 4 Motores DC, 2 Motores de Passo ou 2 Servos; Tensão de saída: 4,5-36V; Corrente de saída: 600mA por canal; Até 4 motores DC bi-direcional com seleção individual de velocidades de 8 bits (cerca de 0,5% de resolução); Até 2 Motores de Passo (Unipolar ou Bipolar) com bobina única, dupla ou passos interlaçados; 4 Pontes H: 0,6A por Ponte (1,2A de pico) com proteção térmica e diodos de proteção contra retro-alimentação; Resistores Pull Down mantem motores desativados durante a entrada de alimentação; Botão de Reset Arduino; Terminais em bloco de 2 pinos e jumper para conexão de alimentação externa. O Motor Shield pode ser observado na Figura 3.

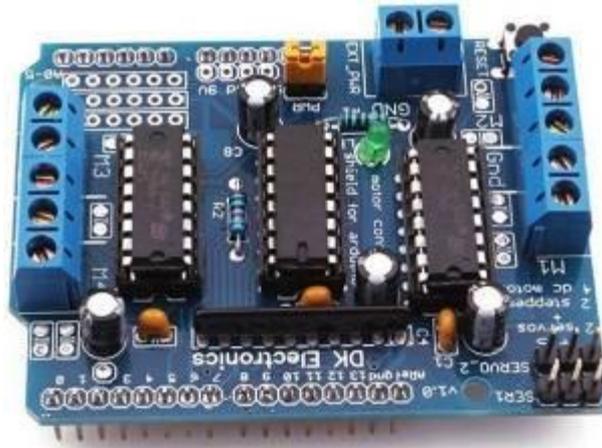


Figura 3. Motor Shield L293D Driver Ponte H para Arduino.

O esquema da ponte H pode ser representado pela Figura 4. As chaves funcionam de modo alternado. Quando as chaves S1 e S4 estão fechadas, as chaves S2 e S3 estão abertas, fazendo o motor funcionar em um sentido. Enquanto que, quando as chaves S2 e S3 estão fechadas, as chaves S1 e S4 estão abertas, o que faz com que o motor funcione em sentido oposto.

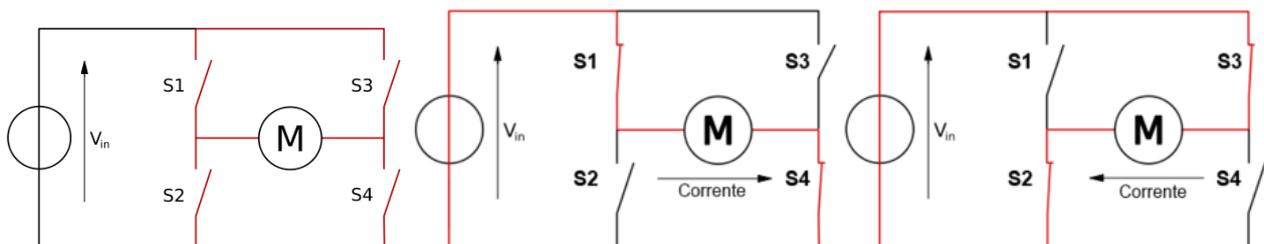


Figura 4. Representação da Ponte H.

O Módulo Bluetooth HC-06 é o meio pela qual é feita a comunicação entre o aplicativo Android e a placa Arduino. Funciona no modo escravo e possui um alcance de até 10 metros. Suas especificações são: Protocolo Bluetooth: v2.0+EDR; Firmware: Linvor 1.8; Frequência: 2,4GHz Banda ISM; Modulação: GFSK; - Emissão de energia: ≤ 4 dBm, Classe 2; Sensibilidade: ≤ -84 dBm com 0,1% BER; Velocidade Assíncrono: 2,1Mbps(Max)/160Kbps; Velocidade Síncrono: 1Mbps/1Mbps; Segurança: Autenticação e Encriptação; Perfil: Porta Serial Bluetooth; Suporta modo Escravo (Slave); CSR chip: Bluetooth v2.0; Banda de Onda: 2,4Hhz-2,8Ghz, Banda ISM;

Tensão: 3,3v (2,7-4.2v); Corrente: Pareado 35mA; Conectado 8mA; Temperatura: -40 ~ +105°C; Alcance: 10m; Baud Rate: 4800; 9600; 19200; 38400; 57600; 115200; 230400; 460800; 921600; 1382400; Dimensões: 26,9 x 13 x 2,2mm; Peso: 9,6g. O Módulo Bluetooth pode ser observado na Figura 4.



Figura 5. Módulo Bluetooth HC-06.

Uma representação da montagem do hardware está exposta na Figura 6. Nela, pode-se observar a ligação do motor 1 ao Motor Shield, representando os 4 motores que são utilizados para movimentar o robô Ziroba em todas as direções. Ainda assim, temos os dois terminais de servo motores (servo 1 e servo 2), que serão utilizados para a ligação da garra robótica. Assim como, o terminal para fonte externa de alimentação.

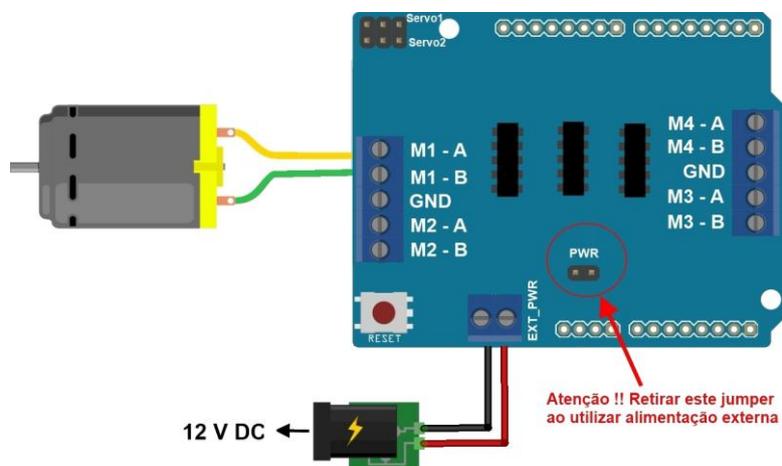


Figura 6. Montagem de hardware.

O código do Arduino foi desenvolvido com o intuito de facilitar ao máximo a sua compreensão utilizando comentários por toda a sua extensão. O mesmo está alicerçado em três partes. A primeira parte, exposta na Figura 7, representa as configurações iniciais de bibliotecas, definições de variáveis dos servos motores, motores DC e inicialização da garra robótica. A segunda parte, representada pela Figura 8, estão expostos os códigos para a verificação constante de comunicação serial entre o aplicativo e o Arduino, de forma a responder aos comandos dados pelo usuário, tais como da movimentação do robô Ziroba (Frente, Tras, Direita, Esquerda e Parar) e da

garra robótica (Descer, Subir, Fecha e Abrir). E, por fim, a terceira parte, ilustrada na Figura 9, representa as funções de controle de movimentação.

```
Projeto_Controlo_Arduino

// Inclusão de bibliotecas para utilização
#include <Servo.h> // Servo Motor
#include <SoftwareSerial.h> // Comunicação Serial
#include <AFMotor.h> // Motor Shield

// Definição de variáveis
SoftwareSerial mySerial(0, 1); // 0-RX, 1-TX
AF_DCMotor motorDC(1); //Seleciona o motor 1
AF_DCMotor motorDC2(2); //Seleciona o motor 2
AF_DCMotor motorDC3(3); //Seleciona o motor 3
AF_DCMotor motorDC4(4); //Seleciona o motor 4
Servo motor; // Seleciona o servo motor 1
Servo motor2; // Seleciona o servo motor 2

void setup(){
  mySerial.begin(9600); // Início da Comunicação Serial

  // Definições de Pinos dos Servo Motores
  motor.attach(10); // Definição do Pino 10 para o servo motor 1
  motor2.attach(9); // Definição do Pino 10 para o servo motor 2

  // Inicialização da Garra
  motor.write(0);
  motor2.write(10);
  delay(10); // Tempo para assentamento dos servo motores
}

```

Figura 7. Primeira parte do Código para Arduino.

```
void loop(){
  // Verificação de caracter para comunicação serial
  char caracter = mySerial.read();

  // Teste condicionais para decisão
  //-----FRENTE-----
  if (caracter == 'F'){
    frente();
  }
  //-----TRAS-----
  if (caracter == 'B'){
    tras();
  }
  //-----ESQUERDA-----
  if (caracter == 'L'){
    esquerda();
  }
  //-----DIREITA-----
  if (caracter == 'R'){
    direita();
  }
  //-----PARAR-----
  if (caracter == 'S'){
    parar();
  }

  //-----GARRA DESCER-----
  if (caracter == 'c'){
    motor.write(80);
    delay(2000);
  }
  //-----GARRA SUBIR-----
  if (caracter == 'v'){
    motor.write(0);
    delay(1000);
  }
  //-----GARRA FECHAR-----
  if (caracter == 'b'){
    motor2.write(40);
    delay(1000);
  }
  //-----GARRA ABRIR-----
  if (caracter == 'n'){
    motor2.write(10);
    delay(2000);
  }
}

```

Figura 8. Segunda parte do Código para Arduino.

```

// FRENTE
void frente() {
  motorDC3.setSpeed(255); //Define a velocidade maxima
  motorDC3.run(FORWARD); //Gira o motor sentido horario
  motorDC4.setSpeed(255);
  motorDC4.run(FORWARD);
  delay(10);
}

// TRAS
void tras() {
  motorDC3.setSpeed(255);
  motorDC3.run(BACKWARD); //Gira o motor sentido antihorario
  motorDC4.setSpeed(255);
  motorDC4.run(BACKWARD);
  delay(10);
}

// ESQUERDA
void esquerda() {
  motorDC3.setSpeed(255);
  motorDC3.run(BACKWARD);
  motorDC4.setSpeed(255);
  motorDC4.run(BACKWARD);
  delay(10);
}

// DIREITA
void direita() {
  motorDC3.setSpeed(255);
  motorDC3.run(FORWARD);
  motorDC4.setSpeed(255);
  motorDC4.run(FORWARD);
  delay(10);
}

// PARAR
void parar() {
  motorDC3.setSpeed(0); //Define a velocidade nula
  motorDC3.run(RELEASE); //Desliga o motor
  motorDC4.setSpeed(0);
  motorDC4.run(RELEASE);
  delay(10);
}

```

Figura 9. Terceira parte do Código para Arduino.

2.2. Plataforma Android – App Inventor

O App Inventor é uma aplicação de código aberto criada e disponibilizada pela empresa Google no ano de 2010 e mantida atualmente pelo Massachusetts Institute of Technology (MIT), localizado nos Estados Unidos da América. A plataforma permite a programadores iniciantes e experientes criarem aplicativos para o sistema operacional Android, utilizando a lógica de programação pegar e arrastar objetos virtuais para desenvolvimento de determinada função. O ambiente de programação é composto por duas seções: o App Inventor Designer, responsável pelo layout e ferramentas utilizadas, exposto na Figura 10, e o App Inventor Blocks Editor, na qual está presente a programação em blocos, representada na Figura 11.

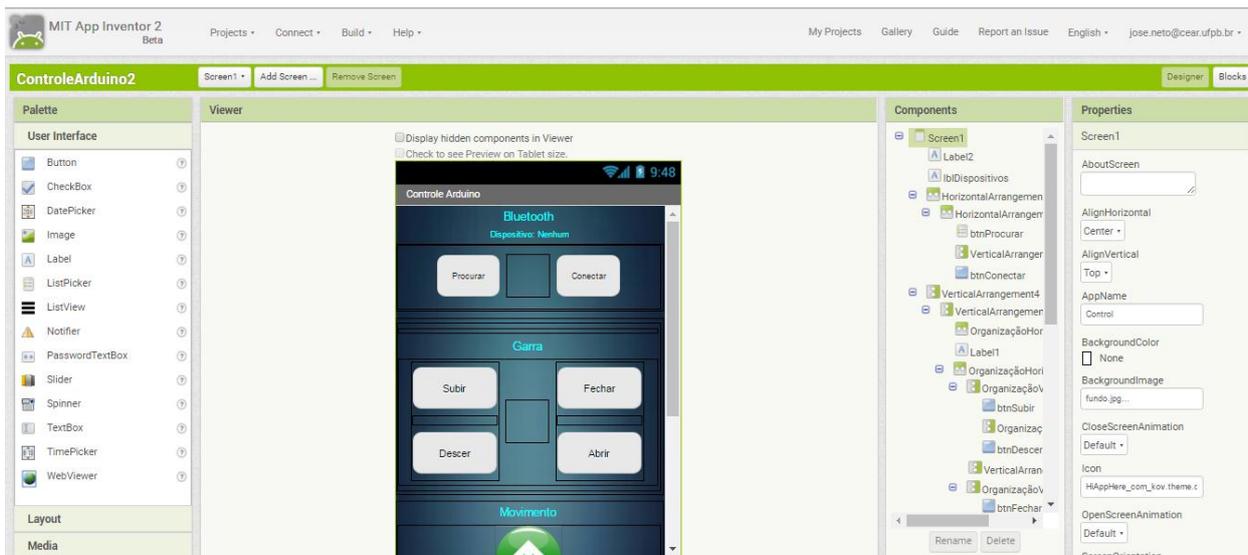


Figura 10. App Inventor Designer.

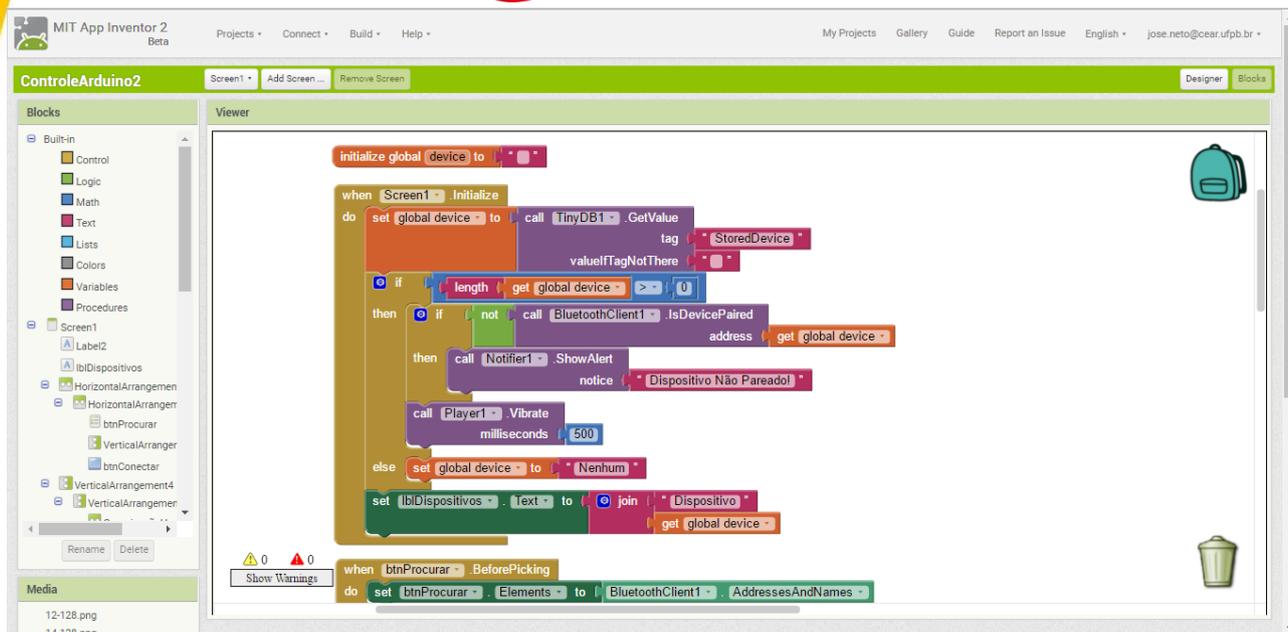


Figura 11. App Inventor Blocks Editor.

Os aplicativos desenvolvidos podem ser analisados de modo instantâneo por emuladores ou baixados e instalados em smartphones para teste. As ferramentas de montagem e o próprio aplicativo são armazenados online durante o seu desenvolvimento no portal: <https://appinventor.mit.edu>.

O app Controle Arduino foi desenvolvido com a proposta de ser de uso intuitivo e com interface simplificada. O mesmo pode ser observado na Figura 12. Na parte superior, há a parte de conexão bluetooth, na qual o usuário pode procurar por dispositivos pareados e conectá-lo ao aplicativo por meio de dois botões. Na parte central, foram utilizados quatro botões para o controle da garra (Subir, Descer, Fechar e Abrir). E, por fim, na parte inferior, há quatro botões para o movimento do robô (Frente, Trás, Direita e Esquerda).

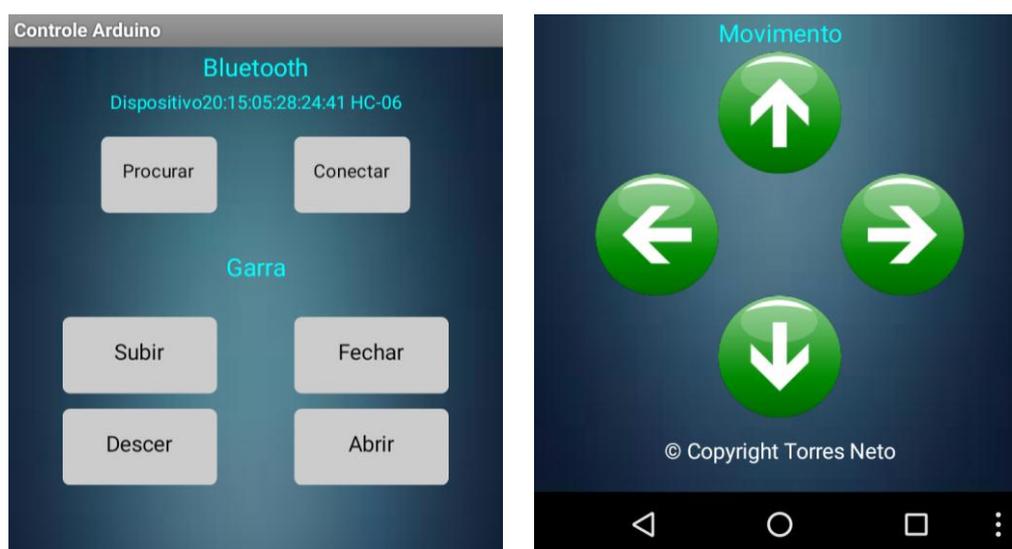


Figura 12. Interface do Aplicativo.

3. RESULTADOS

Como resultado do desenvolvimento do kit de robótica educacional controlado por aplicativo de celular via bluetooth, houve a realização do Desafio de Robótica.

3.1. Desafio de Robótica

Inicialmente, foi realizada uma aula de apresentação do Desafio de Robótica com cerca de 30 (trinta) alunos do 3º ano do Ensino Médio integrado ao Curso Técnico de Geologia no IFPB Campus Picuí. Nessa apresentação, foram expostos conceitos iniciais de robótica, o robô Ziroba e o Aplicativo Controle Arduino. Ao final dessa apresentação, foi disponibilizado o Aplicativo para que os alunos pudessem compartilhar entre si e instalar nos seus smartphones.

Após isso, foi realizado o Desafio de Robótica no Pátio do Campus Picuí. A montagem da pista foi realizada com materiais reciclados, latas de refrigerante, e lixo eletrônico, HD de computador.

As regras foram estabelecidas e anunciadas pelo Professor Fernando Fernandes, Professor da Disciplina de Física do IFPB:

- Os alunos deveriam formar grupos de quatro alunos, nas quais dois participariam de cada bateria;
- A cada bateria, um aluno manipulava o robô, ou seja, colocava-o na posição correta, caso necessário. E o outro, controlava o robô via aplicativo de celular;
- O objetivo do desafio era controlar o robô até a vítima para resgate e leva-la até as coordenadas corretas, sempre utilizando as coordenadas cartesianas, tópico da disciplina curricular de Física;
- A pontuação do grupo foi a soma dos tempos de percurso das duas baterias, mais as penalidades;
- Caso o robô esbarrasse em algum obstáculo, seu tempo foi acrescido em 10 segundos;

A seguir, são apresentadas duas fotos da pista e da execução dos percursos pelos grupos, segundo as Figuras 13 e 14.



Figura 13. Imagem do Desafio de Robótica.



Figura 14. Perspectiva do Desafio de Robótica.

4. DISCUSSÃO

Em pesquisa realizada com os alunos, foi feita uma análise sobre algumas questões referentes ao Desafio de Robótica.

Inicialmente, quanto à organização do Desafio de Robótica, segue o gráfico da Figura 15, todos os alunos, 100%, consideraram excelente.

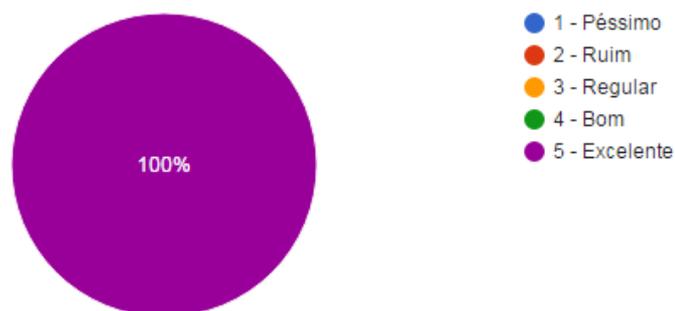


Figura 125. Respostas sobre a organização do Desafio de Robótica.

Quanto ao grau de satisfação com a atividade, conforme o gráfico da Figura 16, dos alunos participantes, 85,7% marcaram a qualificação excelente, enquanto 14,3%, bom.

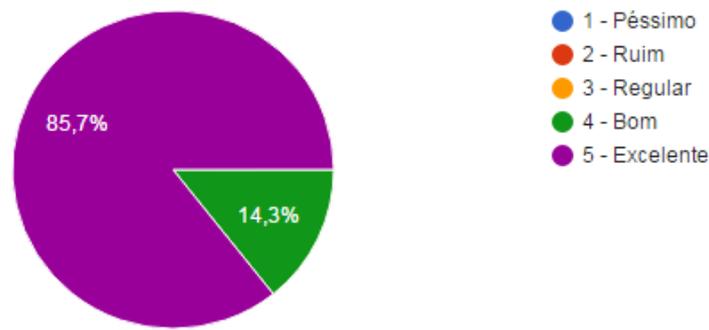


Figura 16. Respostas sobre o grau de satisfação com o Desafio de Robótica.

E, por fim, os alunos responderam se indicariam a realização da atividade para outras pessoas, que no caso, 100% dos alunos disseram que sim, conforme pode ser observado no gráfico da Figura 17.



Figura 17. Respostas sobre a indicação do Desafio de Robótica para outros alunos.

5. CONCLUSÕES

O projeto, desde a sua gênese, buscou o desenvolvimento de uma plataforma simples e acessível aos usuários. Desde custos baixos, utilizando materiais de fácil acesso, até simples programação. Dessa forma, pretende-se disseminar tais práticas no meio acadêmico a fim de ofertar para os alunos vivências extras sala de sala, discutindo temas que são importantes em disciplinas curriculares como a Física.

O projeto se mostrou bastante eficaz quanto à sua aplicação como atividade extra, visto que na avaliação dos alunos, a atividade foi bem avaliada na totalidade dos seus quesitos, conforme pesquisa exposta anteriormente.

Como proposta para execução futura, existe a elaboração de tutoriais para a construção passo-a-passo da plataforma de robótica educacional baseada na placa de desenvolvimento Arduino, no robô Ziroba e no Aplicativo Controle Arduino desenvolvimento pelo App Inventor, integrados à disciplinas da grade curricular, tanto de ensino médio, como de ensino fundamental.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Censo Escolar da Educação Básica 2013. Ministério da Educação. Brasília – DF, 2014.
- [2] L. C Miranda, F. F. Sampaio, J. A. S Borges. RoboFácil: Especificação e Implementação de um Kit de Robótica para a Realidade Educacional Brasileira. Revista Brasileira de Informática na Educação, Volume 18, Número 3, 2010.
- [3] C. Schons, E. Primaz, G. A. P. Wirth. Introdução a Robótica Educativa na Instituição Escolar para Alunos do Ensino Fundamental da Disciplina de Língua Espanhola através das Novas Tecnologias de Aprendizagem. In Anais do I Workshop de Computação da Região Sul, 2004.
- [4] S. R. Zilli. A Robótica Educacional no Ensino Fundamental: Perspectivas e Prática. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Outubro 2004.