

**UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA – EST**

DANILO DA SILVA CUNHA

**PROTÓTIPO DE BAIXO CUSTO DE UMA UNIDADE DE TRACÇÃO
PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA
EMBARCADO**

Manaus

2022

DANILO DA SILVA CUNHA

**PROTÓTIPO DE BAIXO CUSTO DE UMA UNIDADE DE TRAÇÃO
PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA
EMBARCADO**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentado à banca avaliadora do Curso de Engenharia Eletrônica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Engenheiro em Eletrônica.

Orientador: Fábio de Sousa Cardoso, Dr.

Manaus

2022

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST

Reitor:

André Luiz Nunes Zogahib

Vice-Reitor:

Kátia do Nascimento Couceiro

Diretor da Escola Superior de Tecnologia:

Ingrid Sammyne Gadelha Figueiredo

Coordenador do Curso de Engenharia Eletrônica:

Bruno da Gama Monteiro

Banca Avaliadora composta por:

Prof. Fábio de Sousa Cardoso, Dr (Orientador)

Prof. Angilberto Ferreira Sobrinho, Dr.

Prof. Jozias Parente de Oliveira, Dr

Data da defesa: 21/10/2022.

CIP – Catalogação na Publicação

Cunha, Danilo da Silva

Protótipo de baixo custo de uma unidade de tração para cadeiras de rodas manuais com sistema embarcado / Danilo da Silva Cunha; [orientado por] Fábio de Sousa Cardoso. – Manaus: 2022.

61 p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia Eletrônica). Universidade do Estado do Amazonas, 2022.

1. Baixa mobilidade. 2. Deslocamento. 3. Atmega328. I. Cardoso, Fábio de Sousa.

DANILO DA SILVA CUNHA

**PROTÓTIPO DE BAIXO CUSTO DE UMA UNIDADE DE TRAÇÃO
PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA
EMBARCADO**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Eletrônica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Engenheiro em Eletrônica.

Nota obtida: 9,8 (Nove vírgula oito)

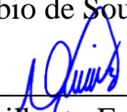
Aprovada em 21 / 10 / 2022.

Área de concentração: Sistemas Embarcados

BANCA EXAMINADORA



Orientador: Fábio de Sousa Cardoso, Dr.


Avaliador: Angilberto Ferreira Sobrinho, Dr.


Avaliador: Jozias Parente de Oliveira, Dr.

Manaus 2022

DEDICATÓRIA

Dedico este trabalho aos meus pais, Ewaldo e Dagmar, à minha irmã, à minha esposa Michelle, aos meus filhos Pedro e Sofia e amigos por todo o amor, carinho, ajuda, paciência e confiança que recebi.

AGRADECIMENTOS

Deus, por me proporcionar a vida, ao meu orientador, pela disposição e mostrar qual caminho seguir. Aos meus pais, que sempre me apoiaram e investiram no meu conhecimento. A todos os professores que contribuíram durante os anos da minha formação acadêmica. A minha mãe, irmã, esposa e filhos, por sempre estar ao meu lado e por me manter motivado. E aos meus amigos que me acompanharam na vida acadêmica e que não mediram esforços para me ajudar durante este desafio.

*“Prefira afrontar o mundo servindo à sua consciência,
a afrontar sua consciência para ser agradável ao
mundo!”.*

Humberto de Campos

RESUMO

O presente trabalho teve como objetivo o desenvolvimento de uma unidade de tração para cadeiras de rodas manuais com sistema embarcado. Com um dispositivo mecânico foi possível acoplar à cadeira de rodas uma unidade de tração que possibilitou o deslocamento e com o auxílio de um aplicativo móvel foi possível uma sinalização de pânico e localização do cadeirante. Inicialmente será apresentada a fundamentação teórica para o desenvolvimento deste trabalho: conceito médico e científico de paraplegia, modo de locomoção de um indivíduo com baixa mobilidade, estudo sobre motores, sensores, gerenciamento de energia, microcontrolador atmega328 e redes de comunicação *bluetooth*. Além disso, poderá ser enviada a localização para um parente ou amigo e acompanhar a rota via aplicativo. Posteriormente, são apresentadas as etapas e materiais necessários à construção da unidade de tração, monitoramento, seguidos da descrição detalhada do experimento, ferramentas utilizadas e implementação do que foi citado. Por fim, os dados obtidos observaram-se os testes de autonomia com o melhor resultado de 37,04 Km, com velocidade média de 10 km/h. O botão de pânico enviou a localização via SMS para o contato cadastrado no aplicativo e a pesquisa de satisfação confirmou que o protótipo desenvolvido atende aos requisitos propostos ao auxiliar o deslocamento de uma pessoa com baixa mobilidade de locomoção em ambientes externos.

Palavras-chave: deslocamento. baixa mobilidade. Atmega328. *bluetooth*.

ABSTRACT

The present work aimed to develop a traction unit for manual wheelchairs with an embedded system. With a mechanical device it was possible to attach a traction unit to the wheelchair that made it possible to move and with the help of a mobile application it was possible to signal panic and locate the wheelchair user. Initially, the theoretical basis for the development of this work will be presented: medical and scientific concept of paraplegia, locomotion of an individual with low mobility, study on motors, sensors, energy management, atmega328 microcontroller and bluetooth communication networks. In addition, the location can be sent to a relative or friend and follow the route via the app. Subsequently, the steps and materials necessary for the construction of the traction unit, monitoring, are presented, followed by a detailed description of the experiment, tools used and implementation of what was mentioned. Finally, the data obtained were observed in the autonomy tests with the best result of 37.04 km, with an average speed of 10 km/h. The panic button sent the location via SMS to the contact registered in the application and the satisfaction survey confirmed that the prototype developed meets the proposed requirements when helping the displacement of a person with low mobility in outdoor environments.

Keywords: displacement. low mobility. Atmega328. bluetooth.

LISTA DE FIGURAS

Figura 1 – Cadeira de rodas manual .	14
Figura 2 – Motores BLDC com um par de polos e com dois pares de polos.	15
Figura 3 – Motores BLDC de rotor externo.	16
Figura 4 – Princípio de funcionamento do efeito Hall.	17
Figura 5 – Sensores de efeito Hall dentro do estator.	18
Figura 6 – Relação entre as saídas dos sensores hall e as tensões aplicadas em cada fase. ...	19
Figura 7 – Arduino UNO R3.	21
Figura 8 – Módulo transceptor HC-05.	22
Figura 9 – O Diagrama das etapas de implementação do projeto.	24
Figura 10 – Divisor de tensão.	27
Figura 11 – Módulo bluetooth HC-05	28
Figura 12 – Esquema elétrico da placa.	29
Figura 13 – Modelo da PCB sem componentes	30
Figura 14 – Visualização da PCB em 3D	30
Figura 15 – Visualização da PCB com os componentes.	32
Figura 16 – Seleção do modelo de placa, Arduino Uno.	33
Figura 17 – Diagrama de blocos de interação dos componentes do protótipo	34
Figura 18 – Diagrama de blocos de interação dos componentes do protótipo.	35
Figura 19 – Tela inicial de acesso ao App Inventor 2	36
Figura 20 – Tela ambiente Designer.	37
Figura 21 – Tela ambiente blocos de programação.	38
Figura 22 – Bloco de controle inicializar tela.	39
Figura 23 – Bloco de evento clique no botão.	39
Figura 24 – Bloco de supervisorio.	40
Figura 25 – Tensão das baterias em série.	41
Figura 26 – Motor fixado no garfo dianteiro.	42
Figura 27 – Controlador motor BLDC.	42
Figura 28 – Acelerador modelo Twist.	43
Figura 29 – Conjunto motorizado, vista frontal.	43
Figura 30 – Conjunto motorizado, vista lateral.	44
Figura 31 – Protótipo montado.	44
Figura 32 – Instalação do aplicativo.	45
Figura 33 – Telas iniciais do aplicativo.	46
Figura 34 – Telas de alerta.	46
Figura 35 – Gráfico da autonomia para 70 kg.	47
Figura 36 – Gráfico da autonomia para 80 kg	47
Figura 37 – Gráfico da autonomia para 90 kg	48
Figura 38 – Testes do botão de pânico	48
Figura 39 – Pesquisa de validação do protótipo.	51
Figura 40 – Resultado da pesquisa, pergunta 4.	51
Figura 41 – Resultado da pesquisa, pergunta 5.	52

SUMÁRIO

INTRODUÇÃO	11
1 REFERENCIAL TEÓRICO	13
1.1 DEFICIÊNCIA FÍSICA	13
1.1.1 Paraplegia	13
1.2 CADEIRA DE RODAS	14
1.3 MOTORES ELÉTRICOS	14
1.3.1 Motores de corrente contínua sem escovas	15
1.4 SENSOR HALL	17
1.5 CONTROLADOR DO MOTOR.....	18
1.6 FONTE DE ALIMENTAÇÃO.....	20
1.7 MICROCONTROLADOR.....	20
1.7.1 Arduino Uno R3	21
1.8 BLUETOOTH	22
1.9 SOFTWARES	23
1.9.1 Arduino IDE	23
1.9.2 Proteus Design Suite	23
1.9.3 MIT App Inventor 2	23
2 MATERIAIS E MÉTODOS	24
2.1 MÉTODO PROPOSTO.....	24
3 IMPLEMENTAÇÃO DO PROJETO	27
3.1 PROJETO DE PCB	27
3.2 MONTAGEM DOS COMPONENTES NA PCB.....	31
3.3 ELABORAÇÃO DO FIRMWARE	32
3.3.1 Estrutura do <i>firmeware</i> e validação	34
3.4 ELABORAÇÃO DO APLICATIVO	36
3.4.1 Ambiente de designer	37
3.4.2 Blocos de programação	37
3.5 VALIDAÇÃO E MONTAGEM DO PROTÓTIPO.....	40
3.5.1 Validação dos componentes do projeto	40
4 RESULTADOS OBTIDOS	47
4.1 CUSTOS.....	49
4.2 PESQUISA DE VALIDAÇÃO DO PROTÓTIPO	50
CONCLUSÃO	53
REFERÊNCIAS BIBLIOGRAFICAS	54
APÊNDICE A – ESQUEMÁTICO DA PLACA	55
APÊNDICE B – CÓDIGO EM C PARA ARDUÍNO UNO	56
APÊNDICE C – APLICATIVO APP INVENTOR 2	59
APÊNDICE D – PESQUISA DE VALIDAÇÃO DO PROTÓTIPO	60

INTRODUÇÃO

O Brasil é um país considerado um dos maiores em população com deficiência, existem 24,5 milhões de brasileiros com algum tipo de incapacidade, física ou mental. Segundo dados do Instituto Brasileiro de Geografia e Estatística (2010), no Amazonas cerca de 790.647 pessoas possuem alguma deficiência. Em Manaus 461.414 pessoas convivem com algum tipo de deficiência, o que demonstra a necessidade de atenção para este público.

As necessidades especiais são incapacidades do indivíduo de realizar suas próprias atividades, fazendo com que o mesmo se torne dependente de um terceiro. A Paraplegia ou Lesão da Medula Espinhal é um tipo de deficiência ao qual o indivíduo não consegue executar os movimentos com os membros inferiores e se torna dependente de uma cadeira de rodas para se locomover. Em função da força realizada ao utilizar a cadeira de rodas manual, o cadeirante sofre um esforço que pode lesionar os membros superiores (SOUZA, 2018). A crescente conscientização referente as dificuldades enfrentadas pelos portadores de necessidades especiais, estimulam cada vez mais estudos para desenvolvimento de ferramentas e equipamentos. O uso da tecnologia é uma das formas de amenizar o impacto na vida das pessoas que precisam exercer suas atividades na sociedade.

O presente trabalho tem como hipótese a ideia de que é possível desenvolver um protótipo de uma unidade de tração que se acople a uma cadeira de rodas manual, empregando o sensor de efeito Hall associado ao microcontrolador Atmega328, controlador de motor *brushless* e unidades de baterias. Através de uma rede de comunicação *bluetooth*, os dados obtidos são enviados e disponibilizados em um aplicativo de *smartphone*. O objetivo deste trabalho foi desenvolver um protótipo de baixo custo de uma unidade de tração que possa ser acoplada a uma cadeira de rodas manual, utilizando motor de corrente contínua sem escovas, fonte de corrente contínua, módulo microcontrolado Arduino Uno R3, botão mecânico de pânico, controlador do motor e módulo *bluetooth* para conexão com dispositivo móvel de modo a exibir e gerenciar, via aplicativo, o nível de bateria, velocidade do protótipo e estado do botão de pânico.

A justificativa para implementação deste trabalho é que a utilização do protótipo possibilitará transformar uma cadeira de rodas manual em motorizada, com botão de pânico e interação com dispositivos móveis que utilizam sistema operacional *Android*. Podendo ajudar na locomoção de pessoas com baixa mobilidade em ambientes externos.

Para o desenvolvimento deste trabalho, foram utilizados vários conceitos estudados no decorrer acadêmico do curso de Engenharia Eletrônica, Microprocessadores e

Microcontroladores, Máquinas Elétricas, Construção Eletrônica, Eletrônica Aplicada, Sistemas Operacionais Embarcados e de Tempo Real, Redes sem Fio e Eletrônica Aplicada II.

Será feita uma revisão teórica dos seguintes assuntos: Deficiência física; Máquinas Elétricas; sensores analógicos e digitais; Sistemas Embarcados, microcontrolador Atmega328, redes de áreas locais e protocolo IEEE 802.15.1.

Este trabalho está dividido em quatro capítulos, os quais são: referencial teórico, materiais e métodos, implementação do projeto, Protótipo de Baixo Custo de uma Unidade de Tração para Cadeiras de Rodas Manuais, com sistema embarcado e análise dos resultados obtidos.

O primeiro capítulo aborda o referencial teórico, onde são abordados os assuntos supracitados que estão relacionados ao projeto.

No segundo capítulo são mostrados os materiais utilizados para a implementação do trabalho, bem como o método proposto. Serão apresentadas as etapas seguidas na implementação do protótipo de sistema, de forma sequencial, começando pela pesquisa e aprofundamento teórico nos assuntos referenciados, seguido da validação dos sensores, construção do protótipo, desenvolvimento e integração do firmware embarcado com o aplicativo de *smartphone* e, por fim, testes de validação final do protótipo de sistema como um todo.

No terceiro capítulo é apresentada a descrição detalhada do sistema que foi implementado, é abordada a execução dos passos citados no capítulo de métodos. Serão apresentados os módulos que constituem o sistema, são descritas as ferramentas e tecnologias utilizadas para a implementação de cada módulo que constitui este protótipo. Para isso, serão descritos os procedimentos de preparação do ambiente de desenvolvimento e implementação do firmware e integração com o aplicativo e, por fim, validação do protótipo de sistema.

No quarto capítulo, são apresentados os resultados obtidos com a implementação do trabalho proposto. Serão apresentadas as análises dos resultados baseadas no referencial teórico e no conhecimento adquirido durante todo o processo de pesquisa e implementação do projeto.

Por fim, é apresentada a conclusão, que aborda a análise do comportamento do protótipo com base na hipótese em comparação com os resultados obtidos. Serão apresentados os desafios encontrados no decorrer da implementação e sugestões para trabalhos futuros.

1 REFERENCIAL TEÓRICO

Neste capítulo, serão abordados os aspectos teóricos dos assuntos relacionados ao projeto. Para o desenvolvimento do protótipo, foram necessários conhecimentos específicos, tais como, deficiência física, características das cadeiras de rodas, motores elétricos, controlador do motor, sensores, fontes de alimentação, microcontrolador, comunicação *bluetooth* e softwares.

1.1 DEFICIÊNCIA FÍSICA

O Ministério do Trabalho e Emprego define deficiência física como:

Alteração completa ou parcial de um ou mais segmentos do corpo humano, acarretando o comprometimento da função física, apresentando-se sob a forma de paraplegia, paraparesia, monoplegia, monoparesia, tetraplegia, tetraparesia, triplegia, triparesia, hemiplegia, hemiparesia, ostomia, amputação ou ausência de membro, paralisia cerebral, nanismo, membros com deformidade congênita ou adquirida, exceto as deformidades estéticas e as que não produzam dificuldades para o desempenho de funções (BRASIL, 2000, Art. 5º, §1º).

De acordo com o censo demográfico, a média de 45 milhões de pessoas declararam ter ao menos um tipo de deficiência, o que corresponde a 24% da população brasileira. Dessas pessoas, aproximadamente 38 milhões estão em áreas urbanas e 7 milhões estão nas áreas rurais (INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, 2010).

Uma das deficiências levantadas no censo é a paraplegia, que se equivale na perda total dos movimentos inferiores do indivíduo.

1.1.1 Paraplegia

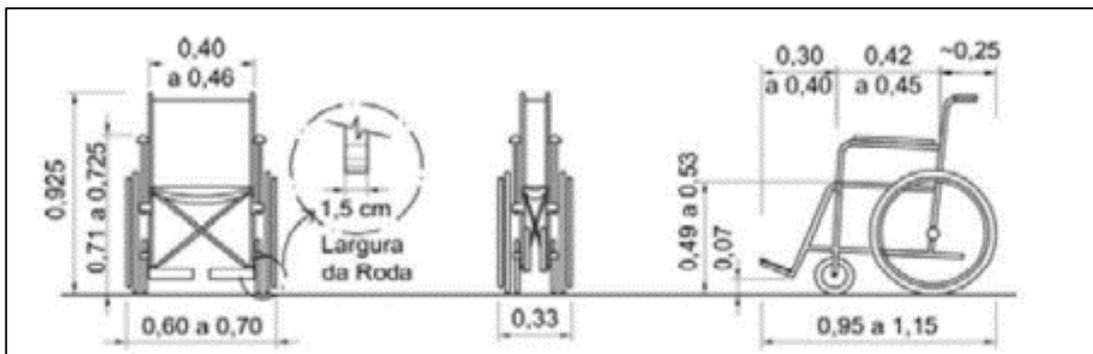
A Paraplegia ou Lesão da Medula Espinhal é qualquer tipo de dano ocasionado na medula espinhal do ser humano, atinge parte do sistema nervoso central que faz a ligação entre os membros superiores e inferiores ao cérebro, através da medula da coluna. É a perda da função motora e ou sensitiva nos segmentos torácicos, lombares e sacrais da medula espinhal. A pessoa paraplégica não consegue executar movimentos com os membros inferiores e em alguns casos membros superiores. Isto pode acontecer por diversas causas, como a má formação do feto, um ferimento por acidente ou um choque elétrico. Essa lesão leva o indivíduo a ser um constante usuário de cadeira de rodas para poder se locomover (MELO, 2012).

1.2 CADEIRA DE RODAS

A cadeira de rodas é um assento montado sobre rodas, seu uso é dado principalmente para pessoas que estão impossibilitadas de andar, ou por algum motivo não movimentam os membros inferiores, isso possibilita que a mesma seja movida manualmente pelo ocupante, eletronicamente por motores ou até mesmo empurrada por alguém. É um dispositivo que fornece mobilidade para pessoas com mobilidade reduzida (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2009).

A cadeira de rodas manual (Figura 1), possui assento, encosto para as costas, descanso para os pés, quatro rodas distribuídas em dois pares, sendo duas dianteiras e duas traseiras que em geral possuem um diâmetro maior que o outro par. A cadeira de rodas manual recebe esse nome devido a sua movimentação ser realizada manualmente pelo usuário ou por um assistente. (SOUZA, 2018).

Figura 1 – Cadeira de rodas manual.



Fonte: (SOUZA, 2018, p. 20).

1.3 MOTORES ELÉTRICOS

Em um sistema motorizado o principal elemento na sua construção é sem dúvida o motor. A escolha desse elemento deve ser feita seguindo os critérios estabelecidos na construção do sistema. O Motor elétrico é uma máquina que transforma energia elétrica em energia mecânica (OLIVEIRA, 2009).

Existem vários tipos de motores elétricos, podendo classificá-los em motores de corrente alternada (CA) e motores de corrente contínua (CC). Motores de corrente alternada funcionam baseados no princípio do campo girante, é necessário que haja vários enrolamentos, em fase, ao redor dos núcleos magnéticos no estator. O motor de corrente contínua funciona

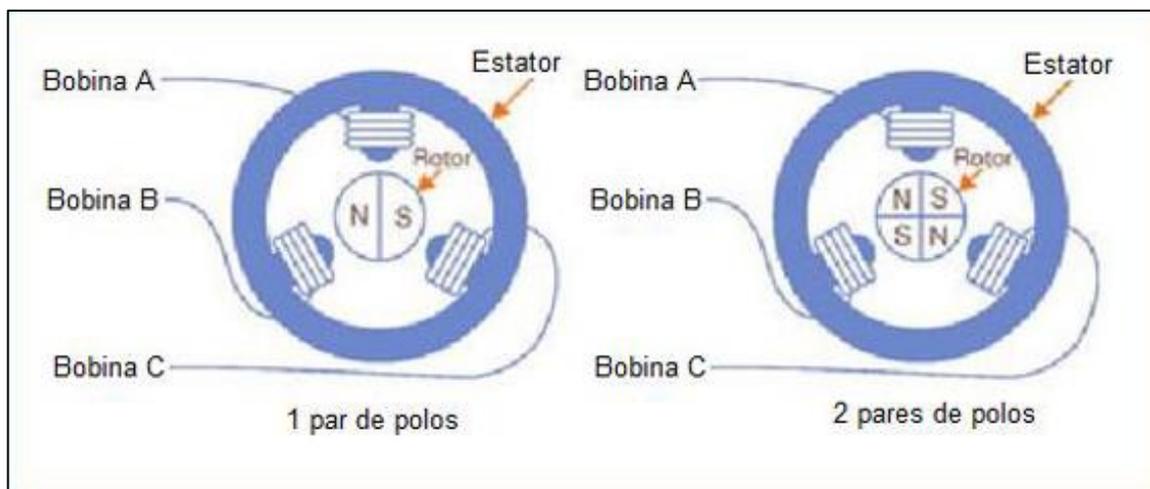
com uma associação de ímãs permanentes e eletroímãs, podendo variar a sua disposição dependendo do tipo de motor (OLIVEIRA, 2009).

Entre os motores elétricos, os motores sem escovas, são os mais indicados para o projeto, devido a sua forma construtiva demandando pouca manutenção e melhor rendimento quando comparados a motores com escovas.

1.3.1 Motores de corrente contínua sem escovas

Um motor de corrente contínua sem escovas, ou no inglês *Brushless DC motor* (BLDC), é um motor que não possui escovas, anéis coletores ou comutadores mecânicos. Os eletroímãs se encontram no estator, e o rotor é feito de ímãs permanentes (Figura 2). O fato de não possuir escovas é uma das vantagens do motor sem escovas sobre o motor com escovas, pois devido ao atrito a que a escova é submetida ela se desgasta e por conta das trocas de conexões das escovas é gerado ruído e em alguns casos, faíscas. Os motores BLDC são síncronos, que significa que o campo magnético gerado pelo estator e o campo magnético formado pelo rotor giram na mesma frequência (GONZALES, 2014).

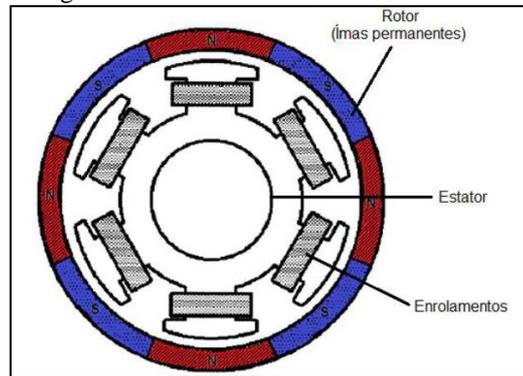
Figura 2 – Motores BLDC com um par de polos e com dois pares de polos.



Fonte: (GONZALES 2014, p. 19).

Os motores BLDC de rotor externo têm seus ímãs permanentes montados na parte móvel da carcaça, que transmite o torque para o eixo. A Figura 3 ilustra a disposição dos elementos desse tipo de motor.

Figura 3 – Motores BLDC de rotor externo.



Fonte: (GONZALES 2014, p. 23).

O rotor é feito de ímãs permanentes de material magnético forte, como o samário de cobalto, ou neodímio de ferro e boro, mas devido ao elevado custo, popularmente é usada a ferrite. Dependendo da aplicação que o motor será empregado o número de polos pode variar; geralmente entre dois e oito polos. A Tabela 1 mostra um comparativo entre motores BLDC e motor CC.

Tabela 1 - Comparativo entre o motor BLDC e o motor CC.

Características	Motor BLDC	Motor CC
Comutações	Comutações eletrônicas baseadas nos sensores de efeito <i>Hall</i>	Comutação das escovas
Manutenção	Baixa manutenção devido à ausência das escovas	Manutenção periódica
Durabilidade	Alta	Baixa
Velocidade/Binário	Permite a operação linear em todas as velocidades	Em velocidades altas o atrito das escovas
Eficiência	Alta, pois não tem queda de tensão nas escovas	Moderada
Inércia do Rotor	Baixa, pois tem ímanes permanentes no rotor.	Alta inercia do rotor.
Ruído Elétrico	Baixo	Alto, devido à geração do arco elétrico nas escovas
Controle	Complexo	Simple

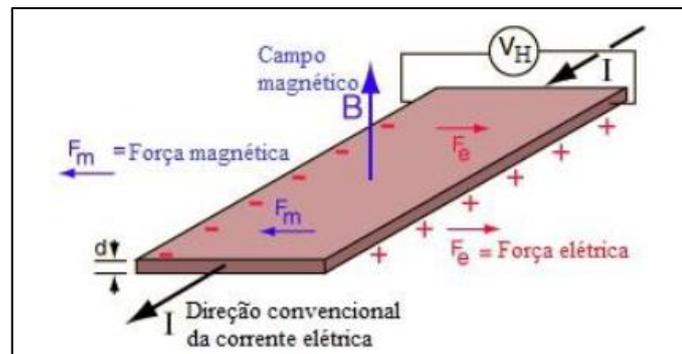
Fonte: adaptado de (GONZALES, 2014).

Para o funcionamento do motor BLDC é necessário de um circuito eletrônico de o controle, pois os enrolamentos devem ser energizados seguindo uma sequência. Faz-se necessário o uso de sensores de posição, para facilitar o controle do motor (GONZALES, 2014).

1.4 SENSOR HALL

Em 1879 o efeito Hall foi descoberto por Edwin H. Hall, que em uma experiência submeteu um condutor elétrico a um campo magnético perpendicular à direção da corrente elétrica, como mostra a Figura 4. Após isso, verificou-se que surgia uma diferença de potencial elétrico nas laterais do condutor na presença desse campo magnético. Este efeito ocorre devido as cargas elétricas tenderem a desviar-se da sua trajetória original por causa da força de Lorentz, criando assim um acumulado de portadores de carga nas superfícies laterais do condutor, onde indica uma diferença de potencial elétrico (ARAUJO, 2012).

Figura 4 – Princípio de funcionamento do efeito Hall.



Fonte: (ARAUJO, 2012).

A corrente atravessa o material semiconductor com os portadores de carga se distribuindo de maneira uniforme, e não é detectada nenhuma tensão no sentido transversal à corrente (I). Porém com um campo magnético (B) externo atuando sobre esse material semiconductor, surge uma força, que será perpendicular ao deslocamento da carga, e faz com que as cargas não atravessem o semiconductor se distribuindo de maneira uniforme, mas sim de maneira modificada, de acordo com a Lei de Lorentz onde demonstra-se que quando uma partícula de carga elétrica q e velocidade \vec{v} está submetida a um campo magnético \vec{B} , é exercida uma força proporcional a q nessa partícula, força essa que é perpendicular à velocidade e ao campo magnético. Temos a equação 1 (GONZALES, 2014).

$$\vec{F} = q\vec{v} \times \vec{B} \quad (\text{Equação 1})$$

A tensão de *Hall* é dada pela equação:

$$V_H = \frac{I.B}{n.e.d} \quad (\text{Equação 2})$$

onde, n é a densidade da mobilidade das cargas, e é a carga do eletro e d é a espessura do condutor.

Após a descoberta do efeito Hall foram desenvolvidos diversos modelos de sensores, e a utilização em motores BLDC é um exemplo. Para este tipo de motores é suficiente obter informação da posição do rotor a cada intervalo de 60 graus elétricos. Esta informação da posição do rotor é posteriormente usada pelo controlador, de forma a executar as comutações do inversor. Nos motores trifásicos é utilizado um sensor de efeito Hall para cada fase (Figura 5), estando estes desfasados de 120 graus entre si, para determinar a posição do campo magnético do rotor. Isto porque sempre que os polos magnéticos passam perto do sensor de efeito Hall é gerado um sinal analógico alto ou baixo indicando a passagem do polo N (norte) ou do polo S (sul) pelo sensor. Normalmente o polo N corresponde ao sinal lógico alto e o polo S ao sinal lógico baixo.

Figura 5 – Sensores de efeito *Hall* dentro do estator.



Fonte: (ARAÚJO, 2012).

1.5 CONTROLADOR DO MOTOR

O motor BLDC exige uma complexidade no seu acionamento, tornando-se necessário um circuito eletrônico para controlá-lo. A forma de controle depende dos aspectos fundamentais do motor, como aceleração, torque ou velocidade máxima (GONZALES, 2014).

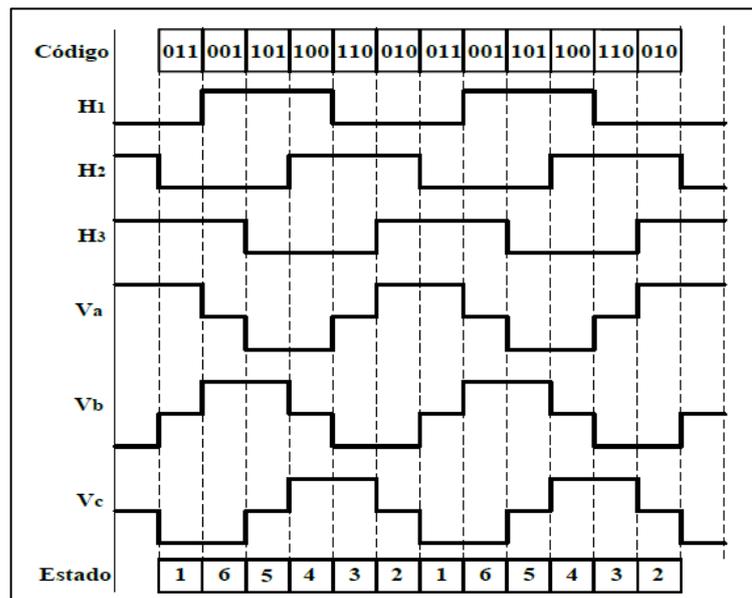
O motor é alimentado em corrente contínua e uma das funções do controlador é comutar essa corrente contínua alimentando as fases do motor sequencialmente. Dentro do motor são fixados os sensores de posição e por sua vez integrados ao controlador, a partir deles o controlador tem a informação necessárias para energizar a fase correta, levando em conta a posição de repouso do rotor, e evitando que ele inicie girando no sentido contrário ao desejado. Também fornecem ao controlador as informações para que o controlador energize as fases do estator em uma sequência correta, de acordo com o sentido e velocidade desejados.

Para o motor BLDC funcionar as bobinas do estator devem ser energizados segundo uma sequência, é fundamental saber a posição rotor do motor, a fim de perceber qual é o enrolamento que deve ser energizado em seguida. A posição do rotor é obtida através de sensores de efeito *Hall* embutidos no estator do motor, a leitura dos sensores é efetuada através da combinação das três saídas digitais dos sensores, onde o valor 0 V corresponde ao sinal digital baixo e o valor de 5 V ao sinal digital alto.

Desta forma cada valor equivale a um estado que por sua vez representa a posição instantânea do rotor do motor BLDC, permitindo assim informar o sistema de controle de qual é o enrolamento que necessita de ser energizado em seguida para fazer rodar o rotor. Assim, baseado na combinação dos sinais lógicos dos três sensores de Hall é determinada a sequência exata da comutação (GONZALES, 2014).

Na Figura 6 é apresentado um diagrama temporal que mostra a relação entre as saídas dos sensores de Hall e as tensões aplicadas ao motor BLDC.

Figura 6 – Relação entre as saídas dos sensores *hall* e as tensões aplicadas em cada fase.



Fonte: (ARAUJO, 2012).

Observa-se que a cada 60 graus elétricos, cada sensor de efeito *hall* muda de estado, resultando num total de seis estados para um ciclo elétrico. No entanto um ciclo elétrico pode não corresponder a um ciclo mecânico do rotor, isto porque o número de ciclos elétricos é determinado pelo número de pares de polos do motor (ARAUJO, 2012).

Na modulação de onda quadrada a 120 graus cada semiconductor está ativo durante 120 graus, e em cada modo de operação apenas são ativos dois dos seis semicondutores, existindo um total de seis modos de operação por período, sendo que, cada um tem a duração de 60 graus.

1.6 FONTE DE ALIMENTAÇÃO

A fonte de alimentação em um sistema motorizado representa uma parte fundamental no projeto, pois trata-se de um projeto de automação, alimentado eletricamente. A escolha da bateria deve levar em conta a potência do dispositivo, tensão, corrente de descarga e recarga e tamanho.

A bateria de chumbo-ácido é composta por um ânodo, que é de chumbo esponjoso, pelo cátodo que é feito de dióxido de chumbo, e o eletrólito, que é uma solução de ácido sulfúrico. Cátodo e ânodo são placas retangulares planas, dispostas em paralelo e interligadas alternadamente, com separadores de material não condutor e poroso formando células. Esse conjunto fica dentro de uma cuba, submerso no eletrólito. O conjunto de células ligadas em série determina a tensão nominal da bateria (ATAIDE, 2010).

Além da bateria de chumbo-ácido existe no mercado as baterias de íon-lítio. Possui um ânodo e eletrólito como condutor, o cátodo é metal-óxido e o ânodo consiste de carbono poroso. Durante a descarga elétrica, os íons fluem do ânodo e durante a carga a direção é inversa e os íons fluem do cátodo para o ânodo. As baterias de íon-lítio possuem grandes vantagens em relação as baterias de chumbo-ácido, por exemplo, densidade da energia elevada, autodescarga baixa, manutenção reduzida e causam menor danos ao meio ambiente quando descartadas corretamente. (ATAIDE, 2010).

1.7 MICROCONTROLADOR

O microcontrolador é um componente bastante versátil podendo, através de sua programação, possuir diversas aplicações. O microcontrolador pode realizar desde o controle da velocidade e posição de um guindaste até o controle das lâmpadas decorativas. O modo de operação do microcontrolador irá variar de acordo com a necessidade.

Para a maioria dos sistemas dedicados, o microcontrolador apresenta-se como a solução mais viável em função do baixo custo, facilidade de uso, versatilidade e da enorme aplicabilidade (BERTONHA, 2007).

1.7.1 Arduino Uno R3

O Arduino (Figura 7) é uma plataforma de código aberto (*Open-Source*) criada em 2005 pelo italiano Massimo Banzi com o intuito de contribuir no aprendizado de eletrônica. Com objetivo de criar uma plataforma de baixo custo e com a possibilidade de desenvolvimento de diversos projetos e protótipos no âmbito da eletrônica.

Figura 7 – Arduino UNO R3.



Fonte: ARDUINO (2018).

O Arduino Uno R3 é uma placa baseada no microcontrolador ATmega328. Na Tabela 2 observa-se que tem 14 pinos de entrada/saída digital, sendo que seis deles podem ser usados como saídas PWM (Modulação por Largura de Pulso, no inglês, *Pulse Wide Modulation*), 6 entradas analógicas, um cristal oscilador de 16MHz, uma conexão USB (Porta Serial Universal, no inglês, *Universal Serial Bus*), uma entrada de alimentação uma conexão ICSP (Programação Serial em Circuito, no inglês, *In Circuit Serial Programming*) e um botão de reset (ARDUINO, 2018).

Tabela 2 – Especificações do Arduino UNO.

Microcontrolador	ATmega328
Tensão de operação	5V
Tensão de entrada (recomendada)	7-12V
Tensão de entrada (limit)	6-20V

Pinos digitais I/O	14
Pinos digitais PWM I/O	6
Pinos de entrada analógica	6
Corrente DC por pino I/O	40 mA
Corrente DC para pino 3.3V	50 mA
Flash Memory	32 KB
Flash Memory para Bootloader	0.5 KB
SRAM	2 KB
EEPROM	1 KB
Velocidade do clock	16 MHz

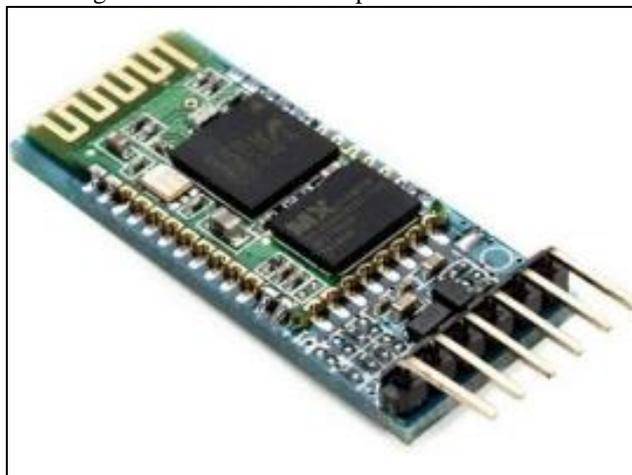
Fonte: ARDUINO (2018).

1.8 BLUETOOTH

O *bluetooth* é um padrão de tecnologia que possibilita comunicação sem fio em pequenas distâncias entre dispositivos eletrônicos a partir de ondas de rádio, criado pela empresa sueca Ericsson em 1994. Definido pelo padrão IEEE 802.15.1 é o primeiro protocolo WPAN (Rede pessoal sem fio, no inglês, *Wireless Personal Area Network*) a se tornar popular comercialmente (OLIVEIRA, 2017).

Será implementado essa tecnologia por meio do módulo *bluetooth* HC-05 (Figura 8) que irá conectar ao Arduino Uno R3 por meio da interfase USART e se conectar ao dispositivo móvel através de uma transmissão sem fio por meio de radiofrequência.

Figura 8 – Módulo transceptor HC-05.



Fonte: (OLIVEIRA, 2017).

1.9 SOFTWARES

No desenvolvimento do projeto foram necessários programas que permitam realizar atividades específicas do projeto, serão abordados os programas, *Arduino Software*, *Proteus Design Suite* e MIT App Inventor.

1.9.1 Arduino IDE

O Arduino IDE é uma plataforma de ambiente de desenvolvimento integrado que possui código aberto e permite fazer *upload* de código dentro de um ambiente de trabalho em tempo real. Desenvolvido para microcontroladores de diversos fabricantes. Sua biblioteca abrange vários modelos de microcontroladores, como, o Atmega328, ATmega1280, stm32F103X, esp32, NodeMCU.

Será utilizada a versão 1.8.13, possui 5 menus com comandos, *File*, *Edit*, *Sketch*, *Tools e Help*, além do ambiente onde será realizada a programação (ARDUINO, 2018).

1.9.2 Proteus Design Suite

Proteus Design Suite é um software para desenvolver projetos de circuitos elétricos, ele tem o objetivo de testar o circuito projetado por meio de simulação e facilitar a confecção de *layouts* de placas de circuito impresso. Nesse projeto será utilizado a versão 8.8 dos módulos ISIS que proporciona a criação e simulação de *layouts* de circuitos e o módulo ARES para criação de Placas de Circuito Impresso (PCI).

1.9.3 MIT App Inventor 2

O MIT App Inventor 2, também conhecido como App Inventor para Android, é uma aplicação código aberto originalmente criada pela Google, e atualmente mantida pelo *Massachusetts Institute of Technology* (MIT). É um ambiente de desenvolvimento web com programação em blocos e que permite usuários com pouca experiência em programação criar um *software* para o sistema operacional *Android*. Além do editor de código ser em blocos ele possui um amplo modulo para desenvolvimento. Nesse projeto é utilizado a plataforma *web* que é aberta, precisando somente realizar o *login* com uma conta Google.

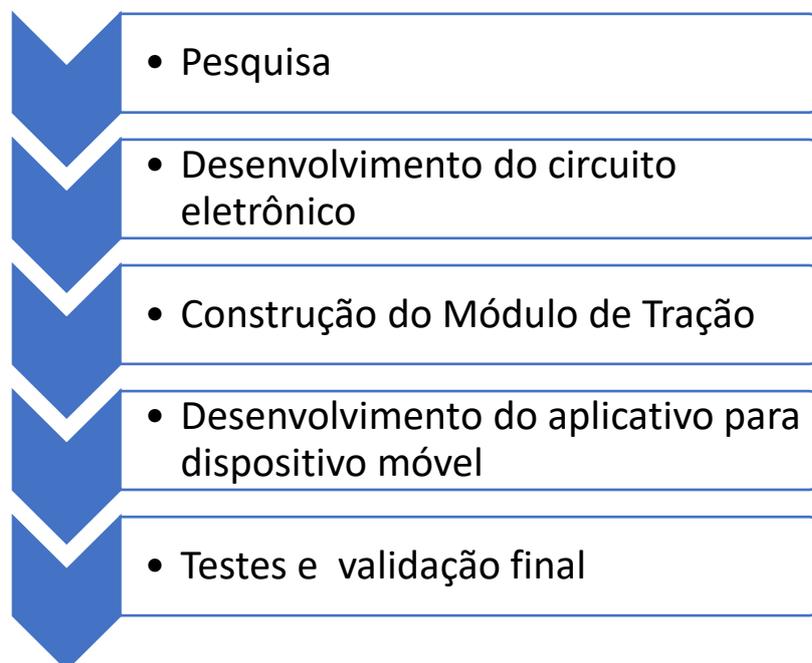
2 MATERIAIS E MÉTODOS

Será apresentado neste capítulo o método proposto, materiais e ferramentas utilizados com as etapas da implementação do projeto.

2.1 MÉTODO PROPOSTO

O método proposto consiste nas seguintes etapas: pesquisa e aprofundamento teórico nos assuntos referenciados, desenvolvimento do circuito eletrônico e circuito impresso, construção do Módulo de Tração, desenvolvimento do aplicativo para dispositivo móvel com sistema operacional *Android 9* e implementação do *firmware* e integração com o aplicativo, por fim, testes de validação final do protótipo de sistema como um todo. Essas etapas podem ser visualizadas na Figura 9.

Figura 9 – Diagrama das etapas de implementação do projeto.



Fonte: Autoria própria.

Na primeira etapa foram realizadas pesquisas na área estrutural e mecânica, de componentes do projeto tendo como referências, modelos já existentes no mercado para realizar

a aquisição dos componentes e elaborar a estrutura e o *hardware* mais adequados para o protótipo, atendendo todos os itens determinados no projeto.

Na segunda etapa, foi elaborado o circuito eletrônico e circuito impresso para interligar o Arduino UNO R3, módulo *bluetooth* HC-05, motor BLDC, bateria e botão de pânico. Foram utilizados nessa etapa os módulos ARES e ISIS do *software Proteus Design Suite* versão 8.8.

Na terceira etapa foi desenvolvido o *firmware*, dando ênfase na programação C e utilizando a interface de desenvolvimento IDE Arduino versão 1.8.16. Foram desenvolvidos os *scripts* de funcionalidade para a realização de leitura do sensor *hall*. Para isso, foram empregadas as bibliotecas *Timer.h* para obtenção do *timer* e *SoftwareSerial.h*. Em seguida, foi implementada a funcionalidade de transmissão e recepção dos dados lidos no sensor e receber comandos por meio de um dispositivo *bluetooth*, *HC-05*, que utiliza o protocolo de comunicação IEEE 802.15.1. Foi realizada a montagem do motor na roda e fixada as partes mecânicas e eletrônicas em um suporte tubular metálico semelhante ao garfo de uma bicicleta e acoplado a cadeira de rodas.

Na quarta etapa, foi criada por meio da ferramenta MIT App Inventor, versão web onde sua IDE é programada em blocos, o aplicativo para dispositivo móvel com sistema operacional *Android 9*. O aplicativo informa para o usuário as informações relevantes, como, nível de bateria, velocidade do conjunto e estado do botão de pânico além das suas coordenadas para quando o botão de pânico for acionado um amigo ou parente poder localizar o cadeirante.

Na quinta e última etapa, foram realizados ajustes finais na montagem das partes mecânicas e eletrônicas, foram realizados testes de validação final. Os dados de entradas foram coletados e os dados de saída foram visualizados na interface do aplicativo em um *smartphone*.

2.2 MATERIAIS UTILIZADOS

Para a confecção do protótipo e testes foram necessários os seguintes materiais e ferramentas:

Proteus Design Suite 8.10 para Windows 10 de 64 *bits*.

Arduino IDE 1.8.13 para Windows 10 de 64 *bits*.

MIT App Inventor 2 para Google Chrome, versão 105.0.51, 64 *bits*.

Ferro de solda de 60 watts.

Fluxo pastoso para soldagem.

Tubo de solda estanho de 1,0 mm.

Alicate de corte.

Tubo termo retrátil 2,0 mm preto.

Tubo termo retrátil 2,5 mm preto.

Fita isolante preta.

1 caixa de derivação cbox-ob 150mm x 110mm x 70mm opaca

1 *protoboard* de 400 pinos.

1 garfo para bicicleta aro 26.

1 aro 26 em alumínio.

1 módulo Arduino UNO R3.

1 módulo *Bluetooth* HC-05.

1 carregador de baterias 48 v

1 multímetro digital Minipa ET-1400.

Cabos para conexões USB.

1 fonte de 5,0 volts a 1,2 ampère.

1 led verde.

1 led vermelho.

3 resistores 1 k Ω (1/8W).

1 resistores 2,2 k Ω (1/8W).

1 resistores 12 k Ω (1/8W).

2 resistores 10 k Ω (1/8W).

4 baterias 12 volts, 7 ah.

1 acelerador tipo *Twist*.

1 botão *push button* vermelho.

1 botão liga desliga.

1 controlador de motor *brushless* 48v, 120 graus.

1 carregador 48 volts.

1 dadeira de rodas simples, 90 Kg.

Cabos para conexões (cabos *jumper*).

3 IMPLEMENTAÇÃO DO PROJETO

Este capítulo apresenta o procedimento detalhado para o desenvolvimento do protótipo, que consiste em um projeto de circuito que interliga o motor, o sensor hall, a bateria, um botão de pânico e um módulo *Bluetooth* ao microcontrolador. Em seguida a elaboração do programa que interpreta a leitura dos sensores no microcontrolador ATmega328 e envia pelo módulo *Bluetooth* HC-05 a um dispositivo móvel. A visualização dos dados será feita em uma interface de visualização de um smartphone, através do aplicativo Android.

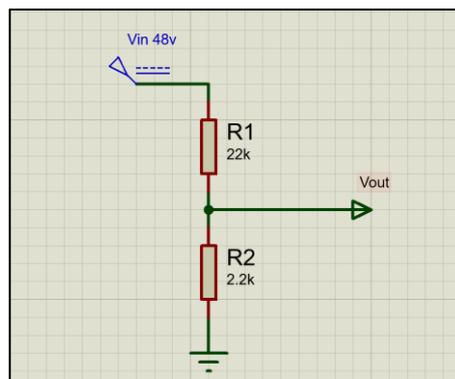
Neste capítulo serão apresentados os seguintes tópicos:

- a) projeto de PCB;
- b) montagem dos componentes na PCB;
- c) elaboração do *firmware*;
- d) elaboração do aplicativo;
- e) validação do protótipo.

3.1 PROJETO DE PCB

Foi utilizado o tamanho e posições dos pinos da placa do Arduíno Uno como base para dimensionar a PCB e que possa se encaixar em cima do Arduíno. Foram utilizadas a porta do Arduino A0, D2, D3, D4, D10 e D11. Para fazer a aquisição da velocidade utilizou-se um sensor indutivo *hall*, S41F, ligado na porta digital D2. Para a medição da autonomia foi montado um divisor de tensão, conforme Figura 10, para adequar a tensão medida num nível de tensão em que a porta do Arduino Uno possa trabalhar sem riscos de queimar.

Figura 10 – Divisor de tensão.



Fonte: Autoria própria.

Foi obtido o valor desejado da tensão a partir do cálculo do divisor de tensão conforme Equação 3

$$V_{out} = V_{in} \left(\frac{R2}{R1+R2} \right) \quad (\text{Equação 3})$$

Para obter a autonomia da bateria foi utilizada a porta analógica A0 com um divisor de tensão que reduziu a tensão de 48 volts para uma tensão que não queimasse a porta. Foi obtido os valores dos resistores R1 e R2 conforme Equação 3.

$$5v = 48v \left(\frac{R2}{R1 + R2} \right)$$

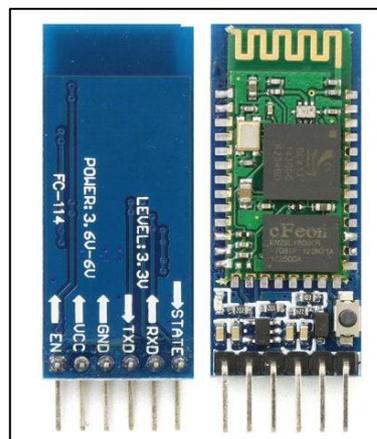
Para R1 = 22k teremos R2 = 2,7k que é o resistor comercial mais próximo. Nesse caso teremos a tensão de 5,24 entrando na porta A0 que é suficiente para proteger o microcontrolador conforme valores de tensão máxima de entrada na porta lógica (Tabela 2).

$$V_{out} = 48v \left(\frac{2.7k}{22k + 2,7k} \right) \quad V_{out} = 5,247$$

No sensor *hall*, e no botão de pânico foram utilizados resistores de *pull-up* de 10k ohms com isso garante-se o nível lógico correto de transição na porta lógica do microcontrolador.

O módulo *bluetooth* HC-05 trabalha com nível de sinal de 3,3 volts, com isso foi necessário um divisor de tensão para limitar a tensão de 5 volts fornecidos pelo Arduino para a tensão de operação de 3.3 volts do módulo HC-05 (Figura 11).

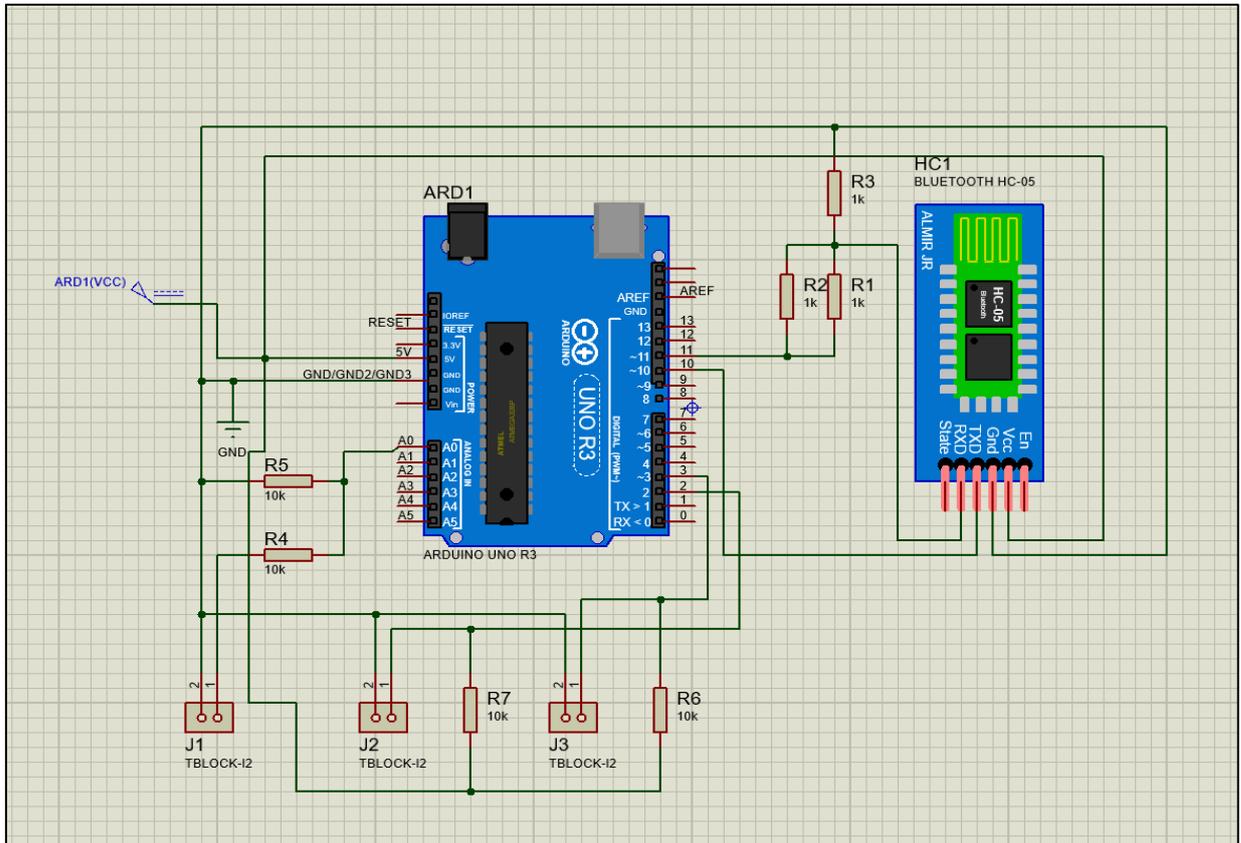
Figura 11 – Módulo *bluetooth* HC-05.



Fonte: Autoria própria.

Foi desenvolvido o esquema elétrico da placa a partir de todas as necessidades do projeto. Com a utilização do *software Proteus Design Suite* versão 8.8, foi desenhado o circuito da placa e a modelagem da placa. A Figura 12 apresenta o circuito elétrico do projeto.

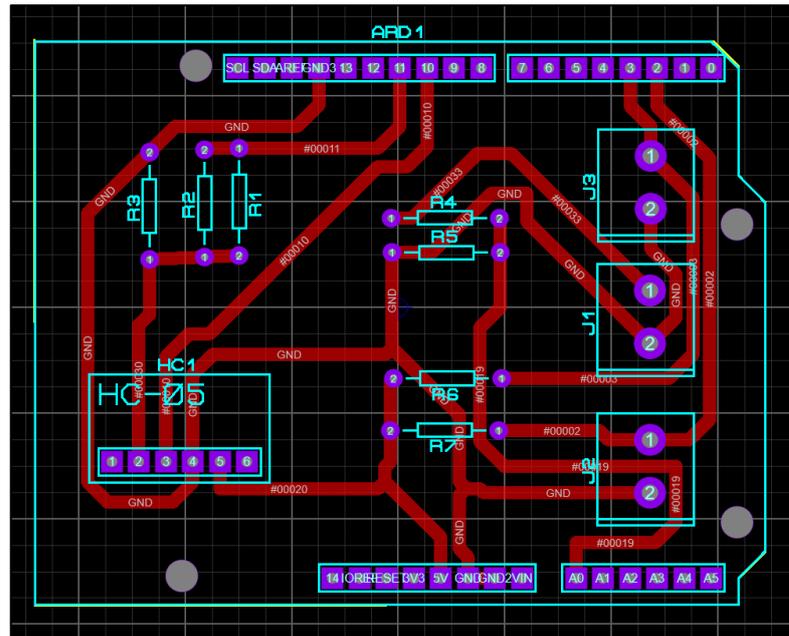
Figura 12 – Esquema elétrico da placa.



Fonte: Autoria própria.

Após elaborado o circuito elétrico o *software Proteus Design Suite* proporciona pelo uso do módulo *PCB Layout* e elaboração do circuito e o roteamento automático das trilhas. A Figura 13 apresenta o *layout* da PCB utilizando a função roteamento automático com a configuração de T50 para as trilhas, o esquema completo pode ser encontrado APÊNDICE A.

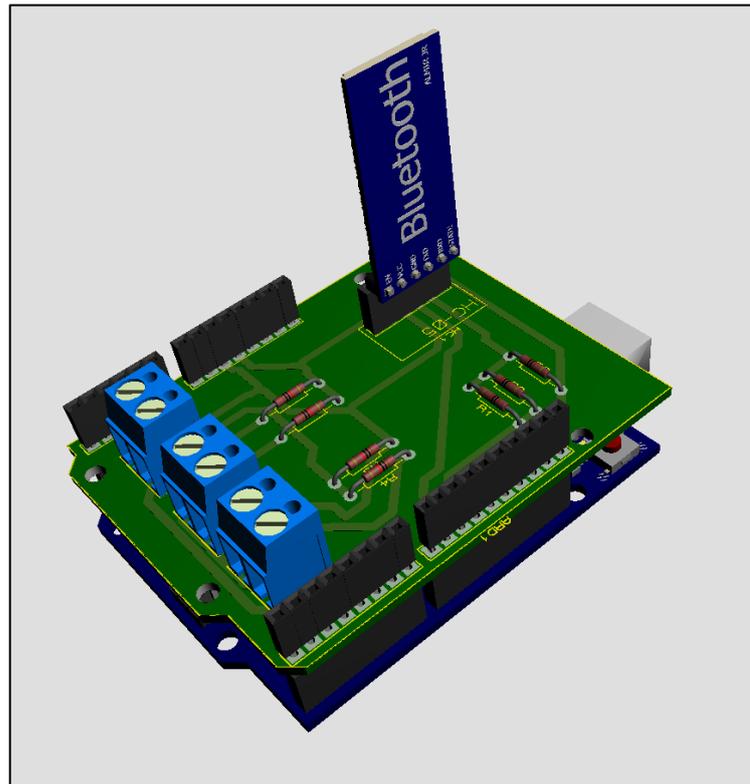
Figura 13 – Modelo da PCB sem componentes.



Fonte: Autoria própria.

O *software* possibilita a visualização 3D do projeto. A Figura 14 apresenta o desenho 3D da PCB.

Figura 14 – Visualização da PCB em 3D.



Fonte: Autoria própria.

Com o layout final definido e os devidos testes realizados, partiu-se para a confecção da placa. A impressão do layout foi feita através de uma impressora a laser em um papel couchê ou fotográfico de gramatura 150 g/m², usa-se a qualidade máxima do toner a fim de evitar falhas. Impressoras de jato de tinta não podem desempenhar essa função.

Com a placa no tamanho determinado no projeto, deve-se retirar toda a gordura desta com uma esponja de aço e detergente ou álcool, não se pode tocar na face limpa novamente.

O desenho é posicionado e fixado sobre o lado cobreado da placa e o conjunto é aquecido, com o auxílio de um ferro de passar roupas em temperatura média, até que todo o toner seja transferido do papel para a placa, o que leva cerca de 15 a 30 minutos. Em seguida, leva-se o conjunto a um recipiente com água, esperando que a tinta solte do papel. E, por fim, retira-se o papel junto com qualquer resquício e, caso haja necessidade, pode-se retocar as falhas com caneta de tinta plástica, caneta de retroprojeter.

Em um recipiente de plástico ou vidro, prepara-se uma solução de corrosivo de perclorato de ferro em água nas proporções indicadas na embalagem, comumente 250ml de água para 100g de perclorato de ferro. A placa deve ficar imersa neste líquido por aproximadamente 15 ou 20 minutos, contudo esse tempo varia com a idade do corrosivo uma vez que este pode ser reutilizado, tendo sua eficácia reduzida. Após o descanso, a placa deve ser lavada com água, sabão e a lã de aço suavemente.

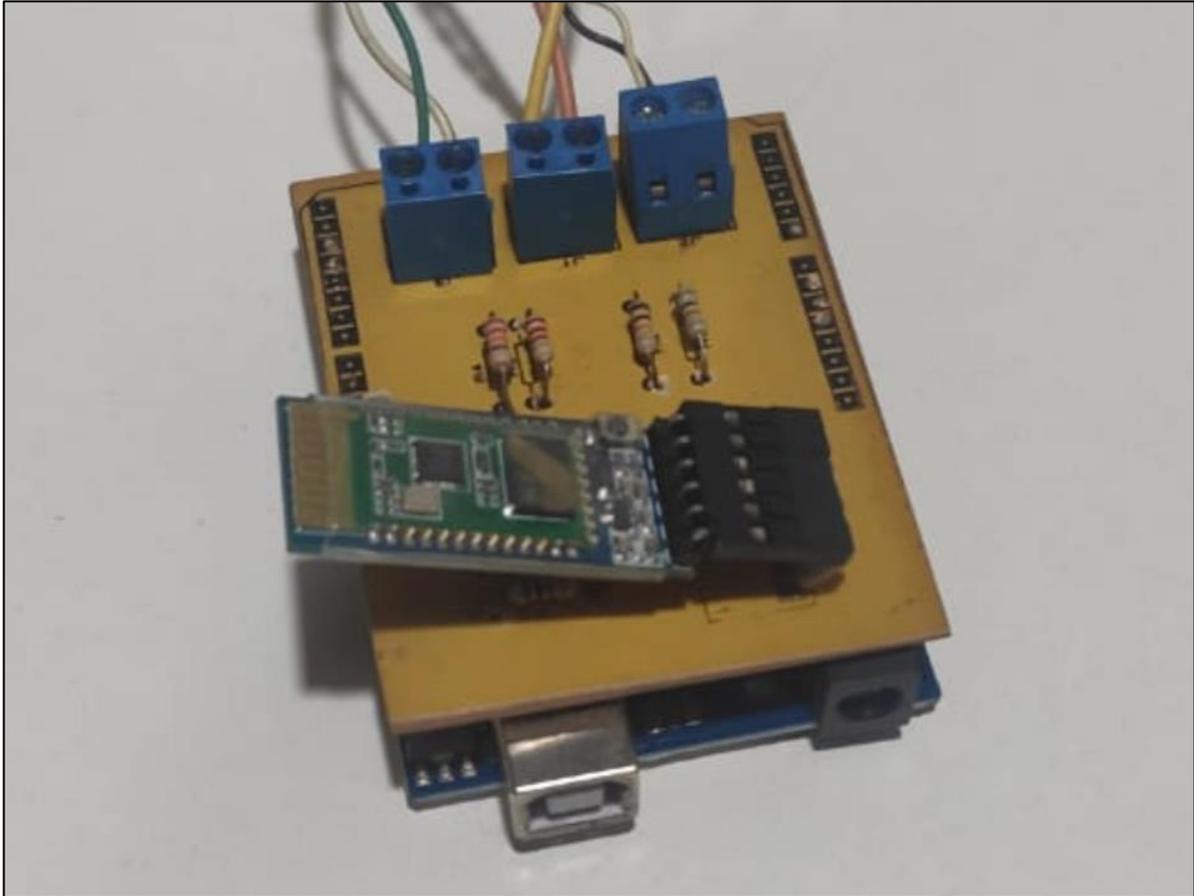
Os centros das ilhas e os furos demarcados nas bordas serão furados com uma furador. Deve-se também lixar as bordas da placa. Pode-se aplicar um produto antioxidante na superfície da placa.

3.2 MONTAGEM DOS COMPONENTES NA PCB

A montagem do circuito na placa. Dispõem-se os componentes nos locais estabelecidos no desenho computacional da board e eles são soldados. Algumas dicas são: manter a ponta do ferro de solda no componente a ser soldado, mantendo a maior superfície de contato possível, e em seguida colocar o estanho; não usar muita solda, usar apenas o necessário; e limpar sempre o ferro de solda, o que vai melhorar a condução.

A PCB pronta com os componentes montados pode ser vista na Figura 15.

Figura 15 – Visualização da PCB com os componentes.



Fonte: Autoria própria.

3.3 ELABORAÇÃO DO FIRMWARE

A escolha do Arduino Uno se deu pelo baixo custo e possui características que se adequam ao projeto. Seu microcontrolador é o ATmega328 de arquitetura avançada RISC AVR da Atmel, possui baixo consumo e otimizado para compiladores C. Nessa placa são dispostos os pinos de entrada e saída, além de disponibilizar uma conexão USB. Essa conexão possibilita a programação, comunicação serial e serve de alimentação para o próprio chip através de um regulador de tensão existente na placa.

Para o projeto foram utilizadas as portas analógicas e as portas digitais do Arduino Uno para obter a leitura dos sensores e atuadores. Deste modo, é apresentado na Tabela 3 a utilização das portas *I/O* e seu respectivo pino da placa do Arduino Uno:

Tabela 3 – Especificações do Arduino UNO.

Utilização	Tipo de porta	Pino
Leitura da tensão da bateria	Analógica	A0
Sensor <i>hall</i>	Digital	2
Botão de pânico	Digital	3
Led	Digital	4
TX – HC-5	Digital	10
RX – HC-05	Digital	11

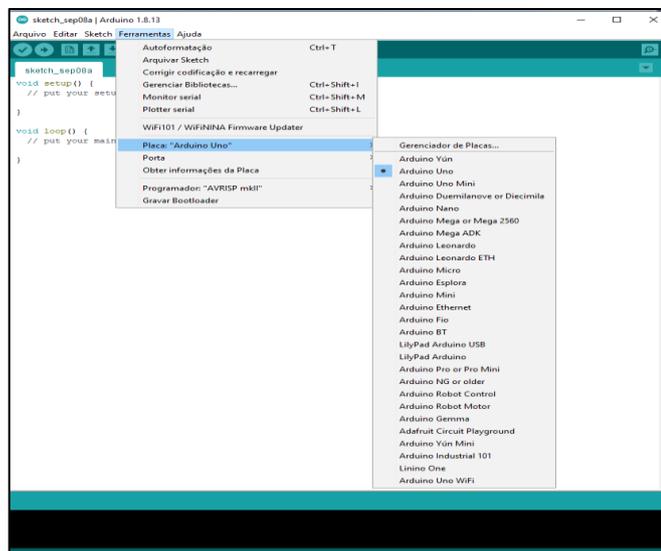
Fonte: Autoria própria.

Para iniciar a implementação, foi necessário configurar o ambiente de desenvolvimento do Arduino IDE (versão 1.8.13). Foram adotadas as seguintes etapas:

- Selecionar o modelo da placa usada, Arduino Uno;
- Configurar a porta serial para monitoramento das portas *I/O*;
- Incluir a biblioteca `SoftwareSerial.h` e `Event.h`;
- Programar utilizando o embasamento teórico dos componentes.
- Observação dos valores das portas *I/O* pelo monitor serial

Após este procedimento (Figura 16), o ambiente de desenvolvimento estava pronto para implementar os códigos e realização de testes.

Figura 16 – Seleção do modelo de placa, Arduino Uno.



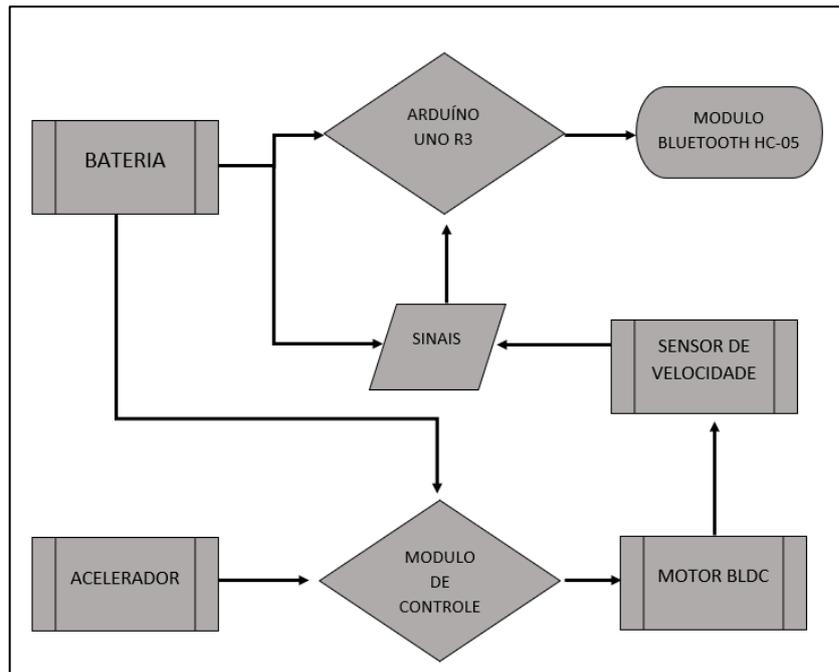
Fonte: Autoria própria.

3.3.1 Estrutura do *firmeware* e validação

O fluxograma (Figura 17), ilustra como o Arduino interage com os outros componentes.

Na IDE do Arduino foram utilizadas as bibliotecas disponíveis, `SoftwareSerial.h` e `Timer.h` para obter o tempo com interrupção e para determinar o tempo de execução do sensor de velocidade.

Figura 17 – Diagrama de blocos de interação dos componentes do protótipo.



Fonte: Autoria própria.

Na autonomia da bateria foi necessária uma implementação de um sensor de tensão, para adquirir o valor instantâneo da tensão nos terminais da bateria. Os sensores de tensão isolados iriam representar um custo adicional ao projeto. Deste modo a medição da tensão é realizada diretamente na porta do Arduino. No entanto foi necessário implementar um divisor resistivo, uma vez que, o valor da tensão da bateria é superior aos valores nominais lidos pela interface com o conversor analógico-digital, do inglês, *Analog Digital Converter (ADC)* do microcontrolador. Desta forma, a tensão que se observa na porta está compreendida entre 0 e 5 V enquanto nos terminais da bateria estarão com uma tensão entre 48 V e 52 V. Com isso observa-se a variação entre a carga máxima da bateria e a mínima, descarregada, e obtém-se a amplitude de variação e determina-se os níveis de tensão percentual da bateria, de 10% a 100%.

O tacômetro informa as rotações por minuto (RPM), enquanto o velocímetro é o instrumento que vai informar a velocidade atual do conjunto. As rotações são obtidas através da comutação dos sensores hall, localizados dentro do motor, que comutam de nível lógico

baixo para alto 23 vezes a cada volta completa. Eles fazem isso devido aos ímãs permanentes localizados dentro do motor.

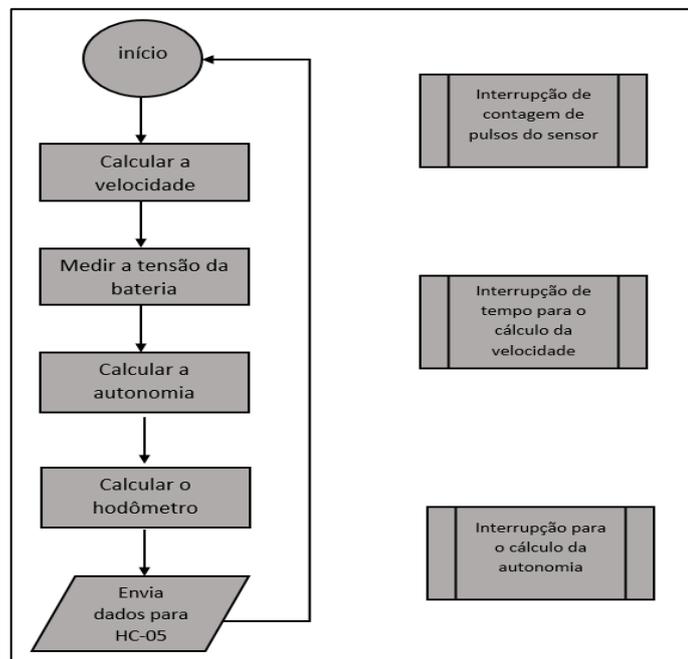
Para cálculo das RPM, é medido o período entre duas comutações e dividido por 23, transformado em rotações por minuto para depois ser calculada a velocidade, conforme equações (4) e (5). O valor de 3,6 é para transformar de m/s para km/h.

$$RPM = \frac{1}{T} * \frac{60}{23} \quad (\text{Equação 4})$$

$$Velocidade_{(km/h)} = 3,6 * RPM * 2 * raio \text{ da roda(m)} \quad (\text{Equação 5})$$

Para a implementação da contagem de quilómetros se utiliza o cálculo do velocímetro. Visto que, a cada volta da roda da unidade de tração é ativada uma interrupção externa, e como em cada interrupção são percorridos aproximadamente uma volta completa da roda, então é contado o número de interrupções externas e multiplicado pelo perímetro da roda. A Figura 18 ilustra o diagrama de interação dos componentes.

Figura 18 – Diagrama de blocos de interação dos componentes do protótipo.



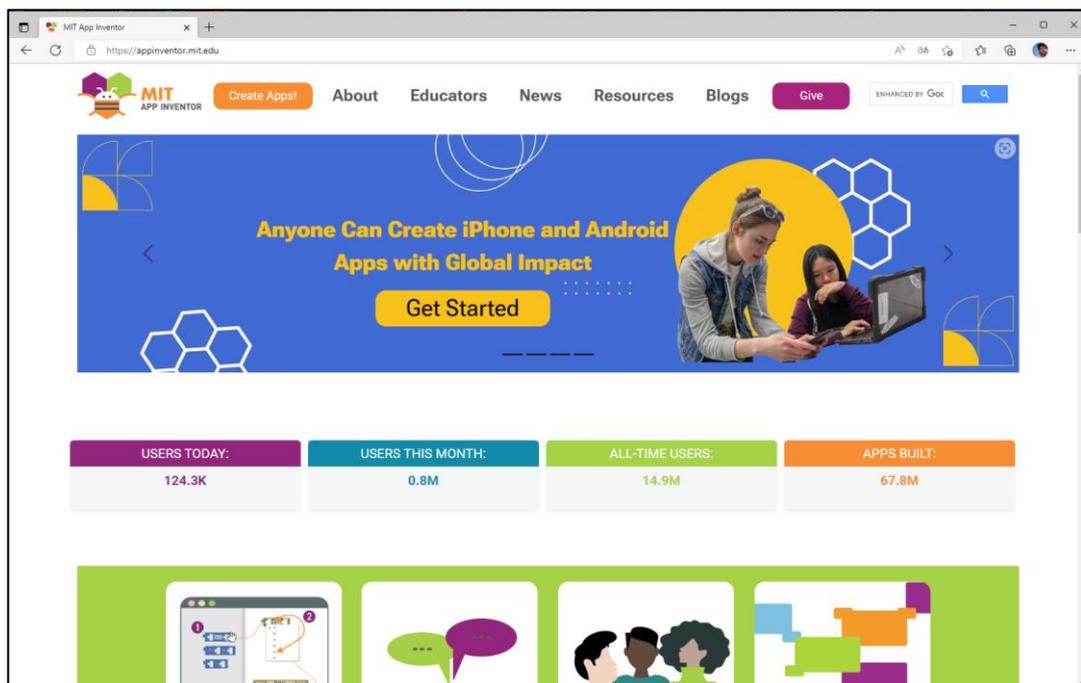
Fonte: Autoria própria.

A programação completa que foi implementada no projeto está disponível no APÊNDICE B.

3.4 ELABORAÇÃO DO APLICATIVO

Para elaborar o aplicativo do *smartphone* Androide foi utilizado a plataforma MIT App Inventor 2 que para acessar basta acessar o link: <<https://appinventor.mit.edu/>> e clicar no ícone: Crie Aplicativos! (Figura 19). O usuário é encaminhado para o endereço eletrônico: <<http://ai2.appinventor.mit.edu/>>, onde é necessário realizar um *login* com uma conta Google.

Figura 19 – Tela inicial de acesso ao App Inventor 2.



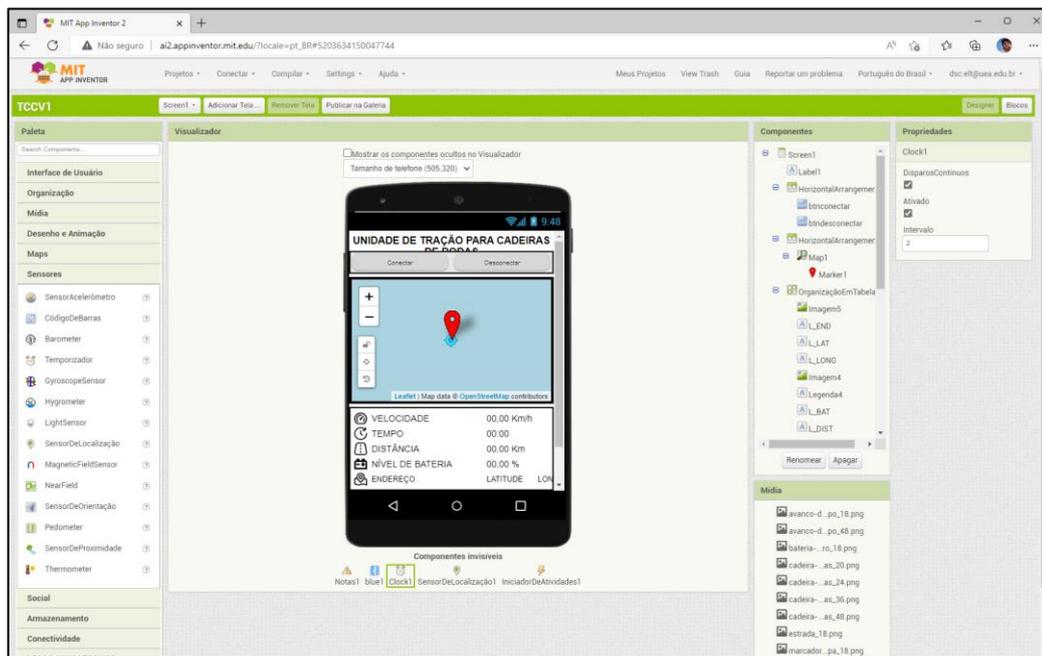
Fonte: Autoria própria.

Os novos aplicativos criados no ambiente App Inventor 2 ficam salvos na própria plataforma, que é online. Eles podem ser acessados a qualquer momento e para que o aplicativo seja utilizado ou simplesmente testado durante o processo de desenvolvimento existem duas possibilidades. A primeira é baixar o aplicativo: MIT AI2 Companion no celular que deve ter o sistema operacional Android. Esse aplicativo é disponível na loja de aplicativos Google Play Store. Ele tem a funcionalidade de ler um código (QR Code) disponível na tela do computador e disponibiliza de forma instantânea o app no celular. A segunda maneira é fazer o uso de um emulador no próprio computador, esta possibilidade não requer a utilização de um celular e simula o funcionamento do app como se fosse no aparelho realmente. A interface do ambiente de programação ocorre de duas maneiras: primeiro por meio de um ambiente de Designer e segundo por um ambiente editor de blocos.

3.4.1 Ambiente de designer

No ambiente de designer (Figura 20) foi configurado o *layout* do aplicativo em desenvolvimento e alguns objetos, como, botões, caixas de texto, imagens, maps, sensores, social, conectividade, *clock* e banco de dados.

Figura 20 – Tela ambiente Designer.



Fonte: Autoria própria.

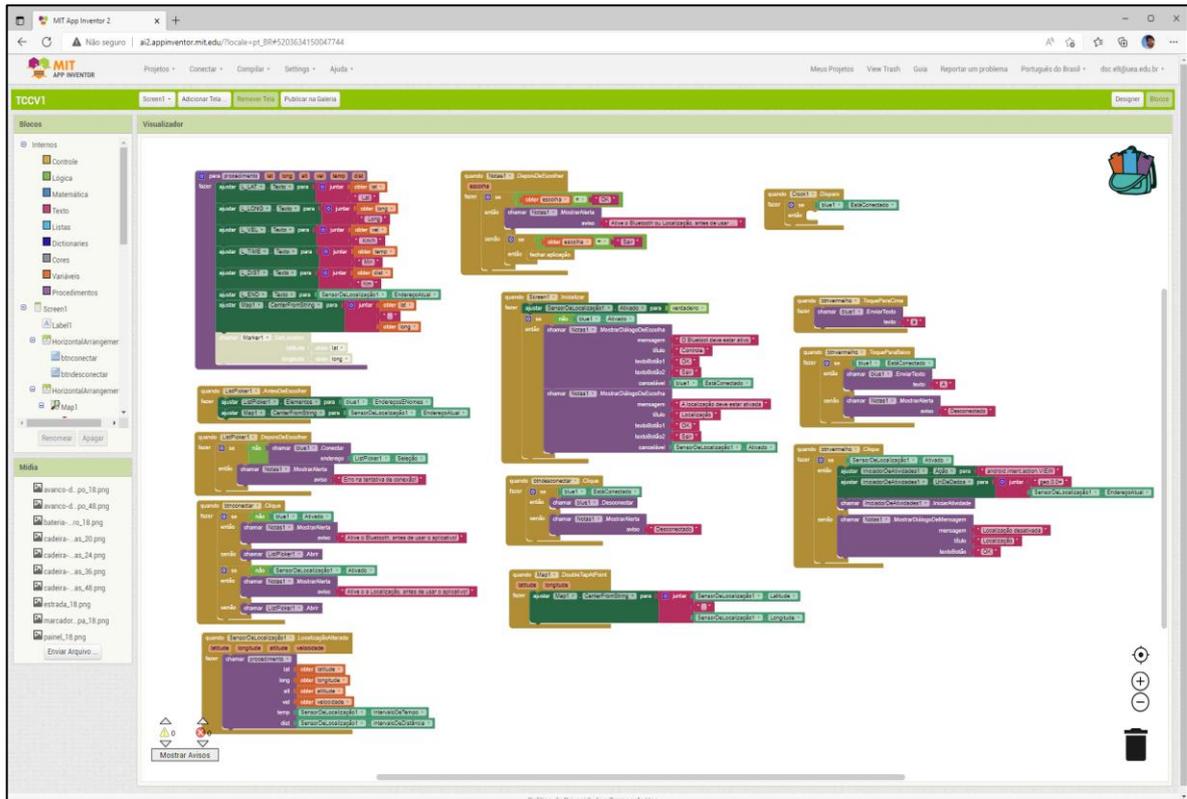
Depois de configurar a parte de visualização do aplicativo, ou seja, a forma como serão apresentadas as interfaces do aplicativo ao usuário, é necessário configurar a parte de blocos. Este ambiente servirá para criar e estruturar os comandos lógicos que permitirão o funcionamento do aplicativo. Aparecerão, também nesta parte, todos os itens inseridos na aba Designer.

3.4.2 Blocos de programação

A programação no MIT é feita como um quebra-cabeça (Figura 21), onde os componentes inseridos no layout do app são encaixados com outros blocos que impõem condições, criadas pelo programador.

Os blocos são utilizados como funções específicas para organizar todos estes itens e os fazer funcionar.

Figura 21 – Tela ambiente blocos de programação.

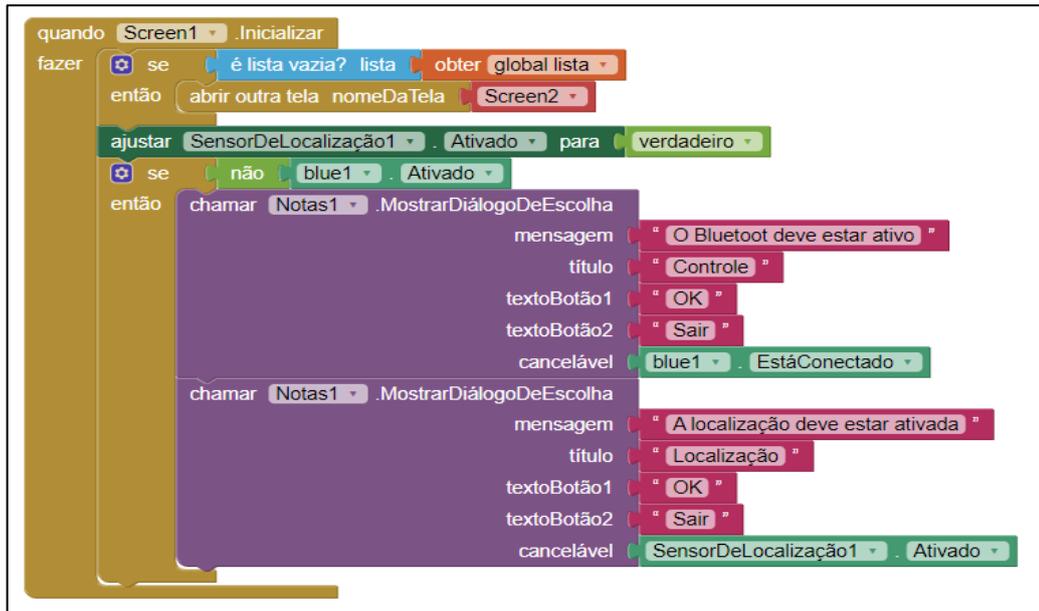


Fonte: Autoria própria.

Depois de configurar a parte de visualização do aplicativo, ou seja, a forma como serão apresentadas as interfaces do aplicativo ao usuário, é necessário configurar a parte de blocos. Este ambiente foi usado para criar e estruturar os comandos lógicos que permitem o funcionamento do aplicativo. Aparecerão, também nesta parte, todos os itens inseridos na aba Designer. Os blocos são utilizados como funções específicas para organizar todos estes itens e os fazer funcionar.

Ao abrir o aplicativo o bloco de controle de inicializar tela, mostrado na Figura 22, inicializa o sensor de localização, verifica se os dispositivos estão conectados e informa o usuário se não tiver ativado. É utilizado as funções de diálogo para alertar se o dispositivo está desconectado ou se o localizador está desligado e se não foram cadastrados um número de contato de emergências.

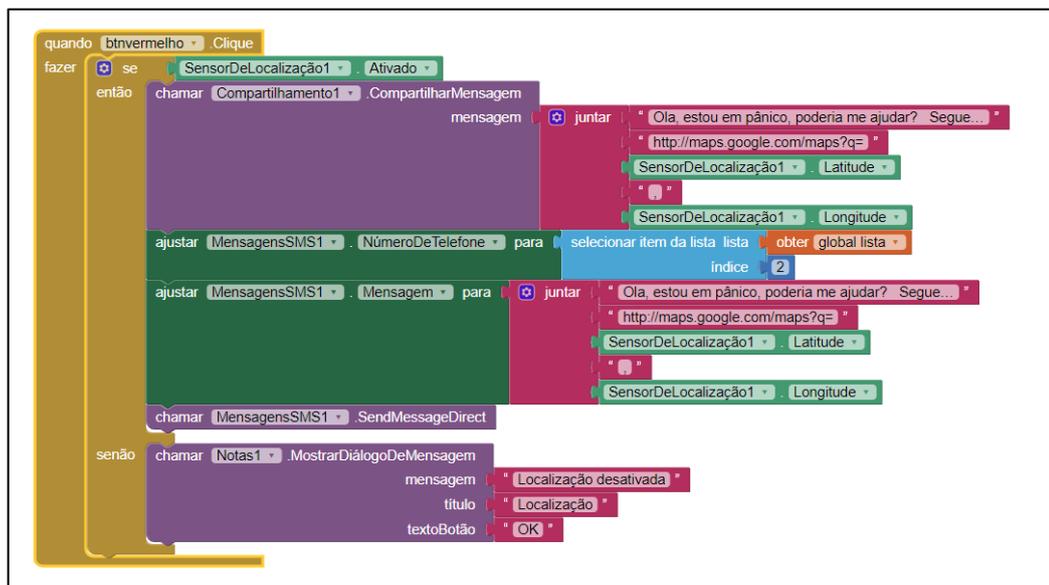
Figura 22 – Bloco de controle inicializar tela.



Fonte: Autoria própria.

Ao clicar no botão, ALERTA, chama-se o evento, clicar no botão, e executa a função de enviar a localização atual via mensagem do inglês *short message servisse (SMS)* para o número cadastrado conforme mostrado na Figura 23.

Figura 23 – Bloco de evento clique no botão.



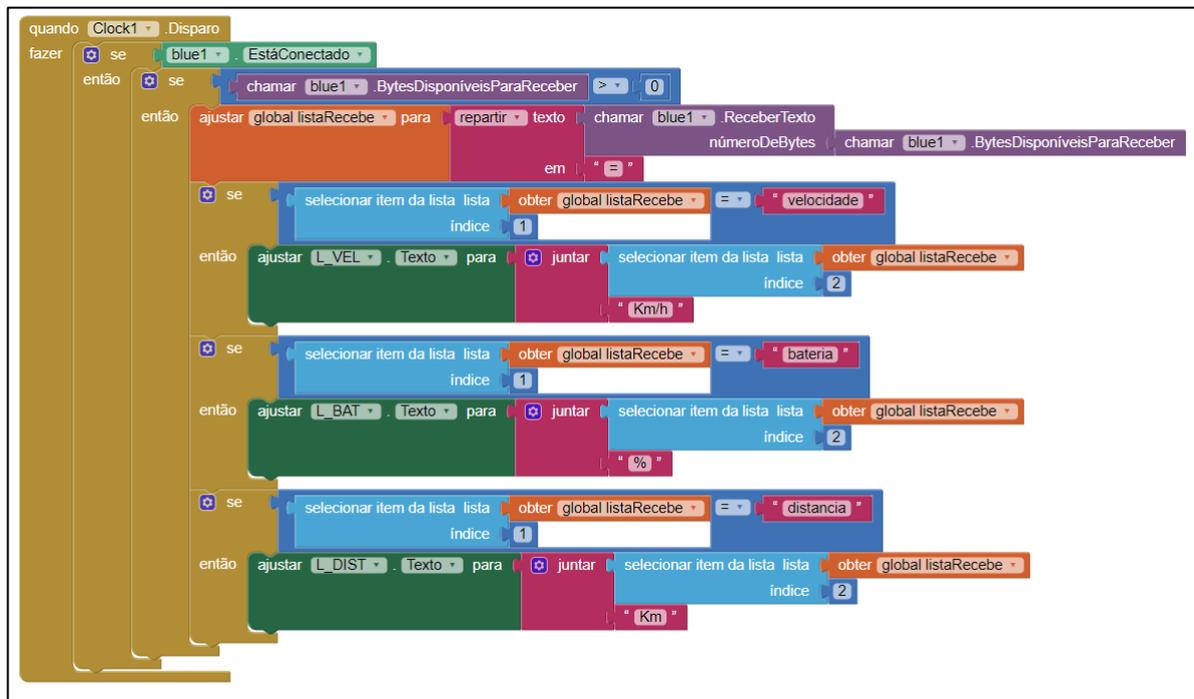
Fonte: Autoria própria.

Foi implementado um supervisor (Figura 24) para receber os dados do *bluetooth*. Ele utiliza a função disparo a cada período e verifica se o *bluetooth* está ativado e se estiver recebe

os dados em formato de lista do dispositivo armazenando em seus respectivos índices as variáveis recebidas.

Mais detalhes sobre a programação em blocos podem ser encontrados no APÊNDICE C.

Figura 24 – Bloco de supervisorio



Fonte: Autoria própria.

3.5 VALIDAÇÃO E MONTAGEM DO PROTÓTIPO

Para iniciar a montagem e validação do protótipo foram feitos os testes em cada componente do projeto.

3.5.1 Validação dos componentes do projeto

Foram utilizadas quatro baterias em série de chumbo-ácido, cada uma com capacidade de 12 V, 7 Ah e foram realizados testes de carga e descarga e aferido o valor resultante do conjunto, conforme mostra Figura 25 obteve-se o valor de 48,4 volts. Foi utilizada uma das baterias para alimentar o módulo Arduino.

Figura 25 – Tensão das baterias em série.



Fonte: Autoria própria.

O carregador de baterias utilizado é do fabricante DYKB. Esse carregador tem as seguintes especificações constantes da Tabela 4:

Tabela 4 – Especificações do carregador de baterias.

Tensão de entrada	100 - 240 V
Frequência de entrada	50 – 60 Hz
Tensão de saída	48 V
Corrente de saída	2 A

Fonte: Autoria própria.

O motor utilizado e acoplado a cadeira de rodas tem, conforme tabela 5, as seguintes especificações:

Tabela 5 – Especificações do motor *brushless*.

Tensão do motor	48 V
Corrente nominal do motor	7,29 A
Potência do motor	350 W
Rotação nominal do motor	547,6 rpm
Frequência nominal do motor	36,4 Hz
Número de polos do motor	23
Resistência do estator por fase	0,453 Ω
Sensor hall	3
Fase	120 °

Fonte: Autoria própria.

Esse tipo de motor é destinado à aplicação em bicicletas elétricas e foi fixado em um garfo metálico conforme Figura 26.

Figura 26 – Motor fixado no garfo dianteiro.



Fonte: (ARAUJO, 2012).

O controlador escolhido (Figura 27) para o projeto seguiu as especificações do fabricante Shen Zhen Rongxing Trading Co., controlador para motores BLDC que trabalha com 36/48 V e 250/350 W, 60/120 ° e pesa 480 g.

Figura 27 – Controlador motor BLDC.



Fonte: Autoria própria.

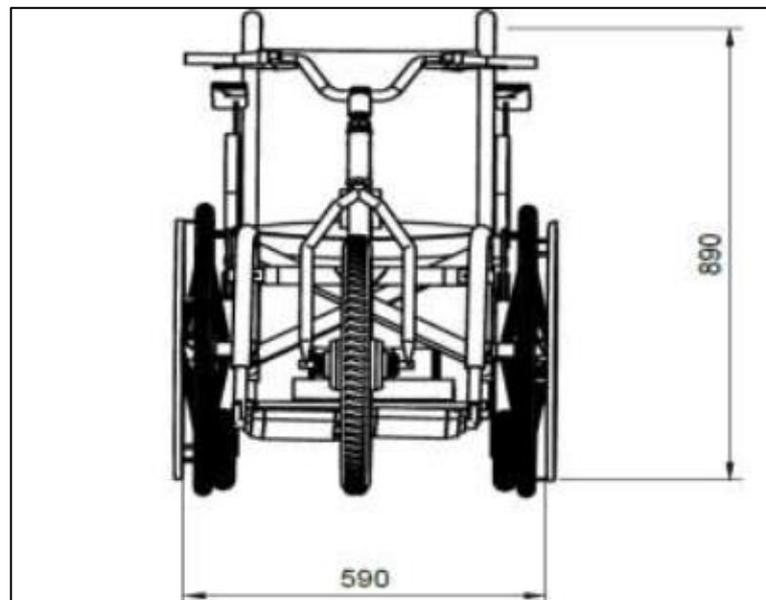
Para controlar a velocidade foi utilizado um acelerador tipo *Twist* (Figura 28), semelhante a um potenciômetro, foi instalado na parte direita do guidão.

Figura 28 – Acelerador modelo *Twist*.

Fonte: Autoria própria.

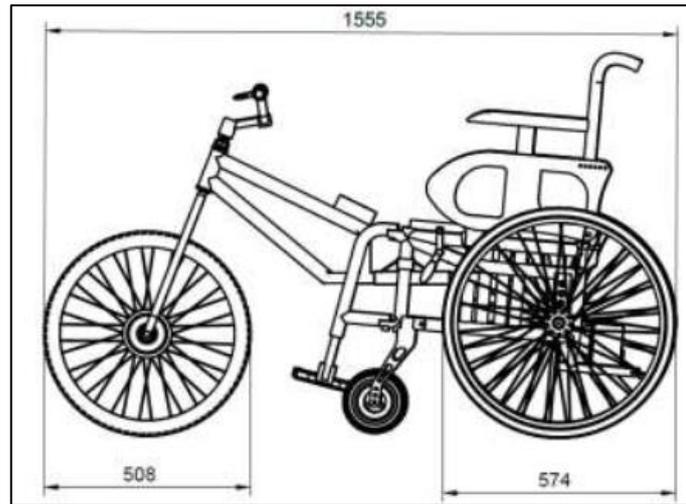
Para o acoplamento da unidade de tração e cadeira de rodas foi utilizado um engate único central que utiliza uma estrutura de ancoragem com uma barra metálica fixada no quadro da cadeira de rodas logo abaixo do acento. E essa estrutura consta com um guia apropriado para realizar o engate. Na unidade de tração se encontra o mecanismo de engate, que é acionado quando está devidamente posicionado na estrutura de ancoragem. A trava é ativada com o usuário empurrando o guidão para frente suspendendo as rodas dianteiras da cadeira. Este tipo de acoplamento é bem simples e rápido. Pode-se observar na Figura 29 a vista frontal do protótipo acoplado a cadeira de rodas e na Figura 30 a vista lateral.

Figura 29 – Conjunto motorizado, vista frontal.



Fonte: (ARAUJO, 2012).

Figura 30 – Conjunto motorizado, vista lateral.



Fonte: (ARAUJO, 2012).

Foi realizada a montagem de todas as partes conforme abordado nos capítulos anteriores e o resultado obtido pode ser visto na Figura 31.

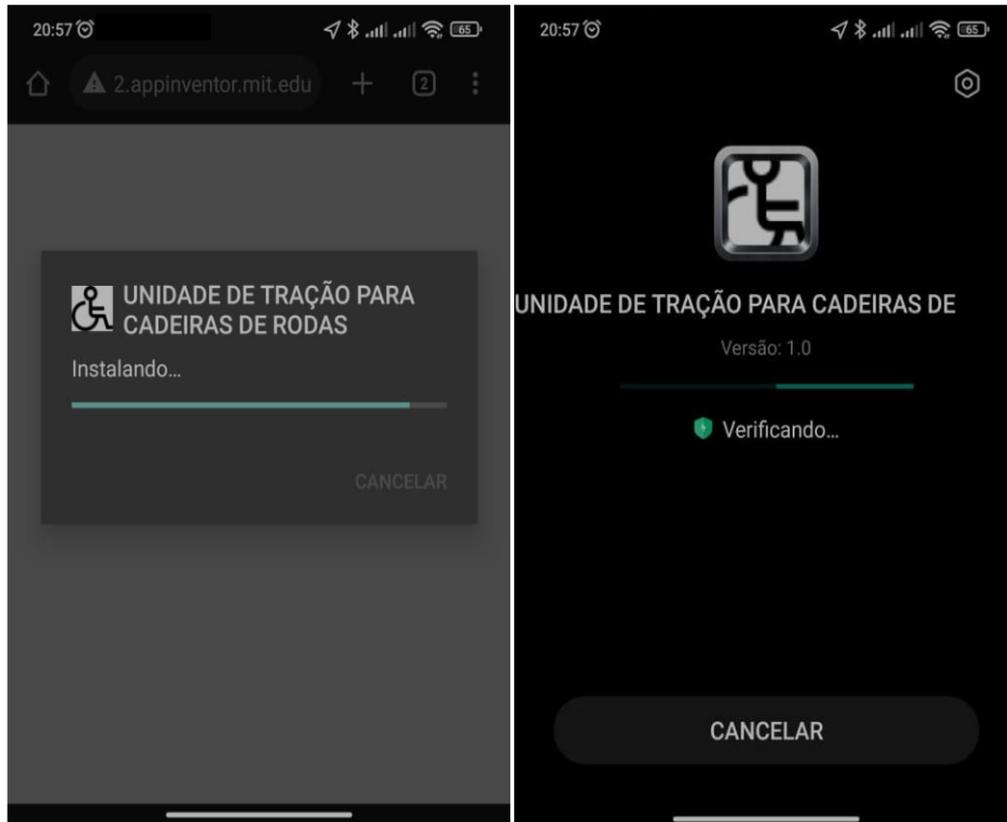
Figura 31 – Protótipo montado.



Fonte: Autoria própria

Após a parte mecânica ser montada foi realizado a instalação do aplicativo no celular com sistema operacional Android versão 11 (Figura 32). O arquivo de instalação está disponível no link: < https://drive.google.com/drive/folders/127jeLX5CqizwV7tZDah1wn-NvIJJ_9w/>.

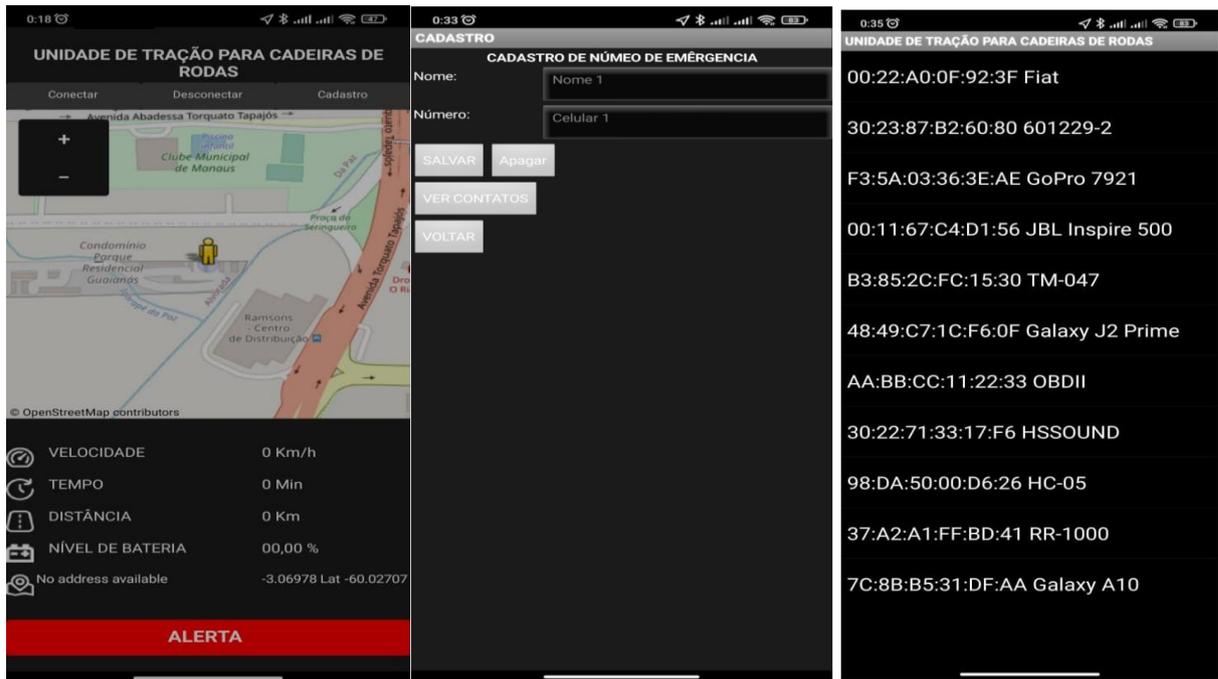
Figura 32 – Instalação do aplicativo.



Fonte: Autoria própria.

Após a instalação e início do aplicativo será mostrado na tela três botões (Figura 33), Conectar, Desconectar e Cadastro, a localização, as informações da unidade de tração e um botão de alerta. Na primeira vez que for aberto, o aplicativo solicita o cadastro com nome e número de contato para uma possível emergência. Após um registro de contato de emergência o usuário poderá se conectar a unidade de tração utilizado o botão conectar. Ao clicar no botão conectar será mostrado uma lista de dispositivos *bluetooth* disponíveis, é selecionado o dispositivo HC-05 que é o módulo *bluetooth* que está na unidade de tração.

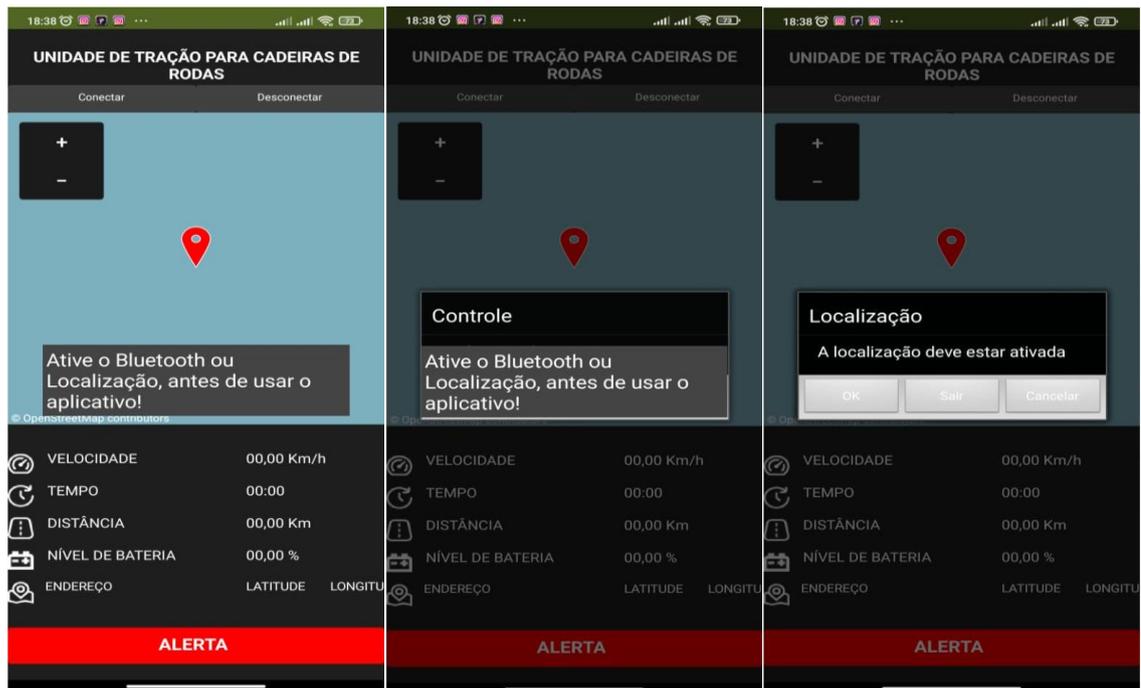
Figura 33 – Telas iniciais do aplicativo.



Fonte: Autoria própria.

Se por algum motivo o *bluetooth* ou o localizador do aparelho celular não estiver ativado será enviado uma notificação ao usuário que para o uso do aplicativo será necessário ativar tanto o *bluetooth* quanto o localizador conforme mostra na Figura 34.

Figura 34 – Telas de alerta.

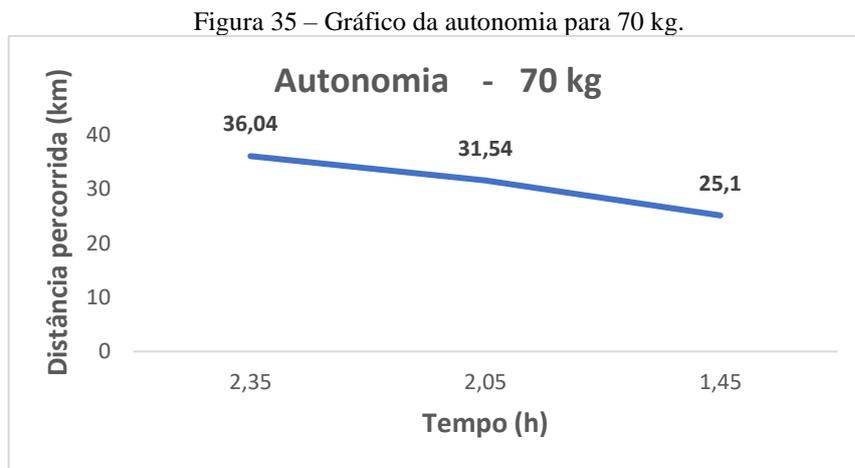


Fonte: Autoria própria.

4 RESULTADOS OBTIDOS

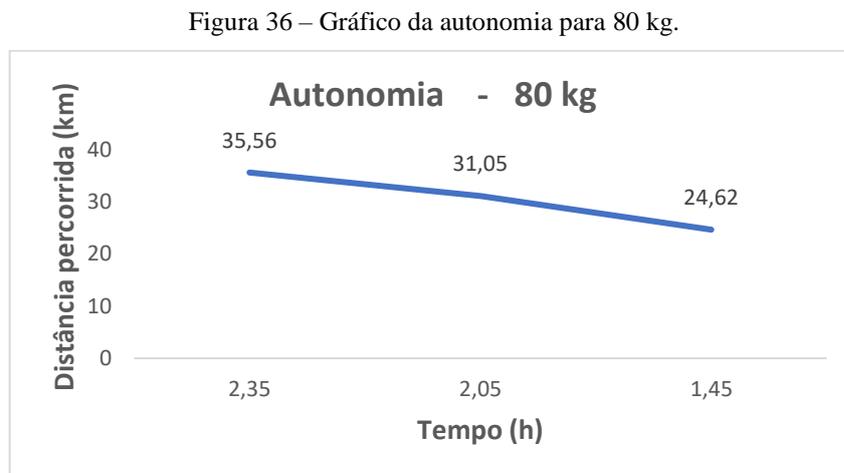
Neste capítulo são apresentados os resultados experimentais da unidade de tração e do aplicativo. Inicialmente são apresentados os testes de autonomia para pesos diferentes e os alertas emitidos pelo aplicativo ao acionar o botão de pânico ou o botão ALERTA no aplicativo.

O primeiro ensaio de autonomia foi realizado com a carga completa das quatro baterias, 48 volts, e com uma pessoa de massa corporal média de 70 kg e velocidade média de 10 Km/h. Foi obtido uma autonomia de 36,04 Km conforme mostra a Figura 35.



Fonte: Autoria própria.

No segundo ensaio foi realizado com a carga completa das quatro baterias, 52 volts, e com uma pessoa de massa corporal média de 80 kg e velocidade média de 10 Km/h. Foi obtido uma autonomia de 35,56 Km conforme mostra a Figura 36.

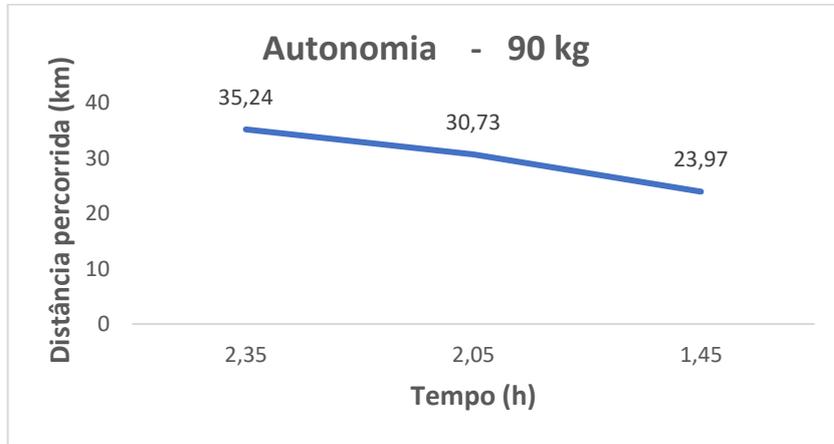


Fonte: Autoria própria.

No terceiro ensaio foi também realizado com a carga completa das quatro baterias, 52

volts, e com uma pessoa de massa corporal média de 90 kg e velocidade média de 10 Km/h. Foi obtido uma autonomia de 35,24 Km conforme mostra a Figura 37.

Figura 37 – Gráfico da autonomia para 90 kg.

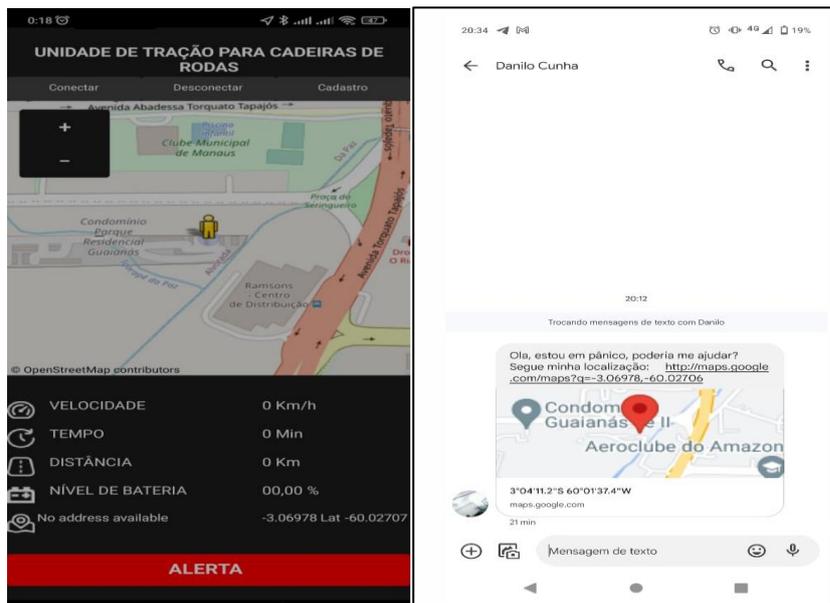


Fonte: Autoria própria.

Os testes foram feitos em uma rua plana e sem obstáculos com temperatura ambiente de 35 ° C e com auxílio da plataforma Google Maps.

Ao pressionar o botão físico ou o botão vermelho, ALERTA (Figura 38), na tela do aplicativo automaticamente foi enviado uma mensagem SMS, OLA, ESTOU EM PÂNICO, PODERIA ME AJUDAR? SEGUE MINHA LOCALIZAÇÃO, para o número cadastrado com a localização atual do cadeirante.

Figura 38 – Testes do botão de pânico



Fonte: Autoria própria.

4.1 CUSTOS

Foram obtidos os custos do projeto com base nos custos primários do protótipo desenvolvido. A Tabela 6 mostra os custos diretos envolvidos na elaboração do projeto.

Tabela 6 – Custos do projeto

Custos do Projeto			
Item	Quantidade	Preço	Total
Dispositivos e Equipamentos			
<i>Kit</i> motorizado (motor, controlador, guidão e manetes)	1	R\$ 670,00	R\$ 670,00
Quadro de bicicleta aço	1	R\$ 80,00	R\$ 80,00
Bateria chumbo-ácido 12V/ 7Ah	4	R\$ 80,00	R\$ 320,00
Demais custos (ferramentas e componentes)			R\$ 150,00
Total em dispositivos e equipamentos			R\$ 1220,00
Serviços			
Adequação do motor na roda aro 26'			R\$ 60,00
Construção do acoplamento mecânico			R\$ 80,00
Demais custos (transporte e pesquisa de fornecedores)			R\$ 90,00
Total em serviços			R\$ 230,00
TOTAL			R\$ 1450,00

Fonte: Autoria própria.

O custo final do protótipo foi de R\$ 1.430,00, sem levar em conta o custo de mão de obra para montar do sistema sob encomenda. O kit WS Liberty vendido pela empresa BIKEMOTO BICICLETAS MOTORIZADAS E ELÉTRICAS que serviu de modelo e base para construção do protótipo é vendido atualmente pelo preço de R\$ 8.600,00, no qual já estão inclusos os custos de mão de obra e montagem, impostos, além do lucro (BIKEMOTO, 2022). A Tabela 7 demonstra as características do projeto desenvolvido e do *kit* WS Liberty, explicitando algumas diferenças entre eles.

Tabela 7 – Comparação de valores, Ws Liberty e o Projeto desenvolvido

Características	WS Liberty	Projeto desenvolvido
Potência do motor	600 W	350 W
Bateria	1 de lítio com 36V/10Ah	4 de chumbo-ácido 12V/ 7Ah
Tamanho da roda (pol)	Aro 20	Aro 26
Carga suportada	150 kg	90 kg
Controlador eletrônico	Sim	Sim
Acelerômetro	Sim	Sim
Material	Aço carbono	Aço carbono
Autonomia	30 km por carga	23 km por carga
Engate <i>one-click</i>	Sim	Não
Preço	R\$ 8600,00	R\$ 1450,00
Peso	13 kg	19 kg
Aplicativo Androide com sistema de pânico.	Não	Sim

Fonte: Autoria própria.

Uma das propostas do projeto, é o desenvolvimento de uma unidade de tração de baixo custo. Pode-se observar após a comparação entre o projeto proposto e o *kit* comercial existe uma diferença sutil nas suas características, porém uma diferença expressiva no valor que chega ser superior a 5 mil reais.

4.2 PESQUISA DE VALIDAÇÃO DO PROTÓTIPO

Foi realizada uma pesquisa qualitativa para validação do projeto pela plataforma Google Forms, A pesquisa completa pode ser observada no APÊNDICE D.

. Foram abordadas 15 pessoas, entre elas, parentes, amigos e pessoas que já utilizam cadeira de rodas. Além da pesquisa foi feito o teste físico no protótipo.

Foram realizadas 5 perguntas conforme Figura 39.

Figura 39 – Pesquisa de validação do protótipo.

PESQUISA TCC
 PROTÓTIPO DE BAIXO CUSTO DE UMA UNIDADE DE TRACÇÃO PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA EMBARCADO

disc.elt@uea.edu.br (não compartilhado) Alternar conta

*Obrigatório

IDADE? *

15-20
 21-30
 31-40
 41-50
 +60

Possui baixa mobilidade ou deficiência? *

Baixa mobilidade.
 Deficiência.

Utiliza cadeira de rodas? *

Elétrica.
 Manual.

Utilizaria uma cadeira de rodas com unidade de tração elétrica com botão de pânico e aplicativo? *

Aplicativo disponível somente para Android.

SIM.
 NÃO.

Grau de satisfação ao usar a UNIDADE DE TRACÇÃO PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA EMBARCADO? *

1 2 3 4 5

NÃO GOSTEI. MUITO SATISFEITO.

Enviar Limpar formulário

Fonte: Autoria própria.

Com o resultado da pesquisa (Figura 40) pode-se observar que 86,7 % dos entrevistados utilizariam o protótipo e 53,3% estava muito satisfeito ao usar o protótipo (Figura 41).

Figura 40 – Resultado da pesquisa, pergunta 4.

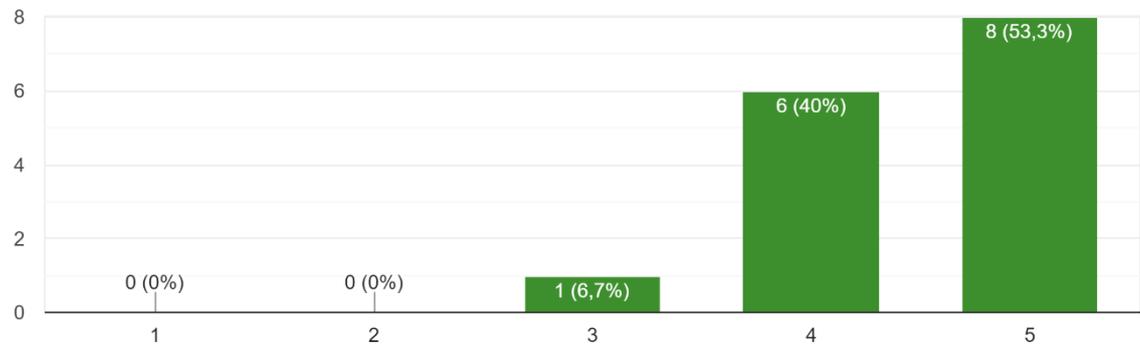


Fonte: Autoria própria.

Figura 41 – Resultado da pesquisa, pergunta 5.

Grau de satisfação ao usar a UNIDADE DE TRAÇÃO PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA EMBARCADO?

15 respostas



Fonte: Autoria própria.

CONCLUSÃO

Diante do grande número de portadores de necessidades especiais que utilizam cadeira de rodas em nosso País, e das reais dificuldades por eles encontradas, faz-se necessária a busca por tecnologia e equipamentos que auxiliem no cotidiano dessas pessoas.

Durante o desenvolvimento dessa pesquisa foram realizados estudos sobre motores elétricos, sensor de campo magnético, aquisição da tensão da bateria. Além disso, foram feitas pesquisas sobre o funcionamento e as principais características dos microcontroladores, mais especificamente sobre o ATmega328. Também foi feito um estudo sobre o protocolo *bluetooth* que possibilita a troca de informações entre a unidade de tração e o dispositivo móvel do cadeirante. No desenvolvimento do projeto observou-se que para implementar o protótipo de uma unidade de tração que se acople a uma cadeira de rodas manual, era necessário o conhecimento de diversas disciplinas cursadas durante a graduação.

Foram apresentadas, de forma detalhada, as etapas de implementação do projeto. Também consta uma apresentação dos *softwares* e ferramentas utilizadas na implementação do referido protótipo

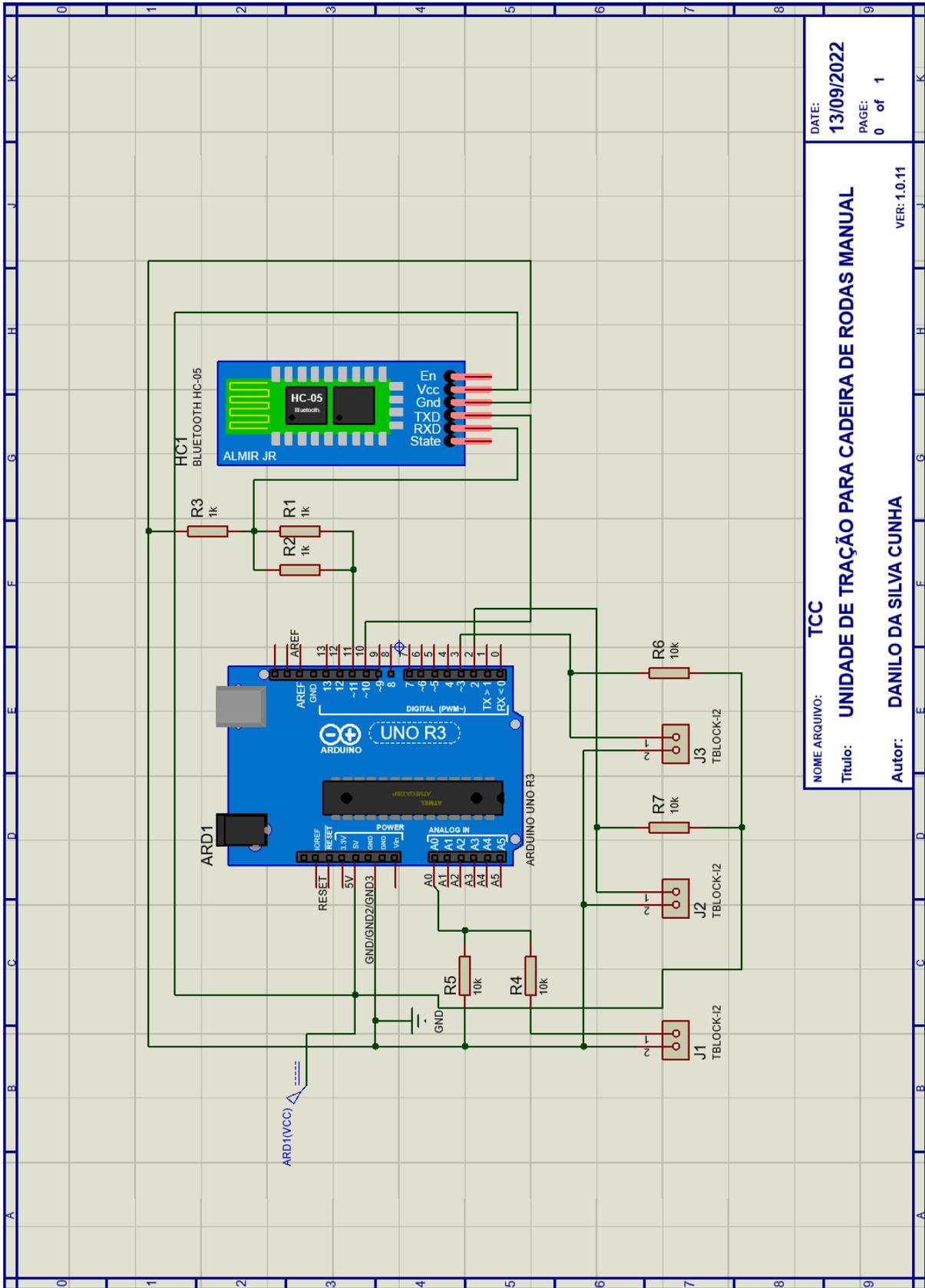
Acerca da hipótese citada na introdução, e com base nos resultados obtidos, pode-se concluir que o protótipo de uma unidade de tração é capaz de se acoplar a uma cadeira de rodas manual e comunicar-se ao dispositivo móvel mandando uma mensagem de pânico com a localização atual a um parente ou amigo quando acionado o botão mecânico ou do aplicativo. Foi verificado, também, que é possível visualizar no aplicativo do *smartphone* o mapa com a localização atual, velocidade média, distância percorrida, tempo percorrido e o nível da bateria.

Para a realização de trabalhos futuros, recomenda-se realizar testes com outro tipo de bateria como as de Li-ion. Recomenda-se elaborar um berço para alojar as baterias com um *design* mais aerodinâmico. Recomenda-se a implementação de função reverso no motor para poder dar ré e a instalação de retrovisor e sinalização visual para o freio, ré e conversões. Recomenda-se um estudo mais profundo no acoplamento de engate único central ou a troca por outro tipo de acoplamento que possibilite uma maior facilidade no engate e não impeça de fechar a cadeira após uso. Recomenda-se também verificar a possibilidade de implementar no aplicativo a função de traçar um destino no mapa e compartilhá-lo e o uso do sensor, acelerômetro, do *smartphone* para identificar colisão ou queda.

REFERÊNCIAS BIBLIOGRÁFICAS

- ARAÚJO, Rui Jorge Fernandes. **Desenvolvimento de uma Bicicleta Elétrica**. Orientador: João Luiz Afonso. 2012. 117 f. Dissertação (Mestrado) - Ciclo de Estudos Integrados Conducentes ao Grau de Mestre, Engenharia Eletrônica Industrial, Universidade do Minho, Azurém, Portugal, 2012. Disponível em <<https://docplayer.com.br/storage/53/30882124/1665845313.pdf>> Acesso em 13 jun. 2021.
- ARDUINO, **Arduino Software**. Arduino, 2018. Disponível em: <<https://www.arduino.cc>>. Acesso em: 2 jun. 2021
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 7176-15**:Normas para cadeiras de rodas. Rio de Janeiro, 2009.
- ATAIDE, Eduardo de Oliveira. **Baterias Recarregáveis**. Campinas: Universidade São Francisco, 2010.
- BERTONHA, Eduardo. **Microcontroladores**. Curitiba: Universidade Tecnológica Federal do Paraná, 2007.
- BIKEMOTO, **Bikemoto**. Bicicletas motorizadas e elétricas, 2022. Disponível em: <https://www.bikemoto.net/KIT_ELETRICO_WSLIBERTY_PARA_CADEIRA_DE_RODAS_600W/prod-2924713/>. Acesso em: 2 jun. 2022
- BRASIL, **Decreto nº 5.296/04**. Brasília: Casa Civil, 2000. Disponível em <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2004/decreto/d5296.htm> Acesso em 13 jun. 2021.
- GONZALES, Vitor Paulino. **Projeto de Uma Bicicleta Elétrica**. Curitiba: Universidade Tecnológica Federal do Paraná, 2014.
- INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, **Censo Demográfico 2010**. [s.l.]: [s.n], 2010. Disponível em: <https://ftp.ibge.gov.br/Censos/Censo_Demografico_2010/Caracteristicas_Gerais_Religiao_Deficiencia/xls/Unidades_da_Federacao/amazonas.zip>. Acesso em: 20 abr. 2021.
- MELO, Wyara Ferreira *et al.* **Lesão da Medula Espinhal (LME) paraplegia**. Mossoró: Revista Verde, 2012. Disponível em: <https://www.gvaa.com.br/revista/index.php/RVADS/article/view/1858/1365>. Acesso em 06 jun. 2021.
- OLIVEIRA, João G. S. M. **Materiais Usados na Construção de Motores Elétricos**. Porto Alegre: Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio Grande do Sul, 2009.
- OLIVEIRA, S. **Internet das Coisas com ESP8266, Arduino e Raspberry Pi**. São Paulo: Novatec, 2017.
- SOUZA, Filipe Augusto. **Adaptação de alavanca de bombeamento á cadeira de roda**. Varginha: Centro Universitário do Sul de Minas, 2018.

APÊNDICE A – ESQUEMÁTICO DA PLACA



NOME ARQUIVO: **TCC**
 Título: **UNIDADE DE TRACÇÃO PARA CADEIRA DE RODAS MANUAL**
 Autor: **DANILO DA SILVA CUNHA**
 DATE: **13/09/2022**
 PAGE: **0 of 1**
 VER: **1.0.11**

APÊNDICE B – CÓDIGO EM C PARA ARDUÍNO UNO

```

/* *****
* Projeto TCC - UNIDADE DE TRACÇÃO PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA
EMBARCADO UNIDADE DE TRACÇÃO PARA CADEIRAS DE RODAS MANUAIS COM SISTEMA
EMBARCADO
*
* DANILO DA SILVA CUNHA
*/

// ##### DECLARAÇÕES #####

#include <SoftwareSerial.h>
#include <Event.h>
#include "Timer.h"

SoftwareSerial serialdoblueetooth(10,11); // ##### Portas para o serial do bluetooth (RX, TX).

#define REFRESH_TIME 1000
Timer t;

int tensao, rpm;
int voltas = 23;
float tensaoCalc;
float tensaoLeitura; // ##### Tensão de leitura da baterias estão carregadas.
volatile float autonomia;
volatile float vel;
const float perimetro = 2.041; // ##### Diametro do aro 26 = 650mm - perimetro = 2pir
float odometro;
float calcOdometro;
volatile long contador =0 ;
volatile long contOdometro =0;
unsigned long lastRefresh;

int valorblue; // ##### Variável que vai ler o valor enviado recebido bluetooth

const int botao=3; // ##### Botao pânico
boolean estadoBotao=false;
int estadobotao=0;

// ##### FUNÇÕES DO PROGRAMA #####

void velocidadeCalc() // ##### Função para contador de velocidade
{
  contador++;
  contOdometro++;
  rpm++;
}

void auTo() // ##### Função para calcular autonomia da bateria.
{

  tensaoLeitura = map(analogRead(A0), 0, 1023, 0, 100);
  autonomia = tensaoLeitura;
  String bateria="bateria=" + String(autonomia)+"=";
  serialdoblueetooth.print(bateria);

}

```

```

void enviaStatus(){ // ##### Função aciona botão

    delay(1000);

    if (estadobotao==!HIGH){
        serialdoblueetooth.print("BOTAO=LIG");
        delay(1000);
    }
    else{
        serialdoblueetooth.print("BOTAO=DES");
        delay(1000);
    } // fim envia status

}

// ##### INICIA O PROGRAMA #####

void setup()
{
    Serial.begin(9600);
    serialdoblueetooth.begin(9600); // ##### Início da serial do bluetooth
    tensaoLeitura = 0;
    autonomia = 0;
    vel = 0;
    odometro = 0;
    contOdometro = 0;
    contador = 0;
    lastRefresh = millis();
    pinMode(3, INPUT_PULLUP); // ##### Configurando a porta do botão de emergência
    pinMode(2, INPUT_PULLUP); // ##### Configurando a porta do sensor Hall
    pinMode(A0, INPUT); // ##### Configurando a porta do sensor de tensão da bateria
    t.every(10000, auTo);
    estadobotao=digitalRead(botao);
    attachInterrupt(digitalPinToInterrupt(2),velocidadeCalc, RISING); // ##### Ativa Interrupção da porta digital 2
    acionada quando o sensor hall muda de um estado lógico para outro.
}

void loop()
{

// ##### Leitura Bluetooth

    if(serialdoblueetooth.available())
    {
        valorblue=serialdoblueetooth.read();
        Serial.print("Valor Recebido por Bluetooth = ");

        if (valorblue=='R'){ // ##### Botão de pânico acionado

            digitalWrite(botao,HIGH);
            delay(1000);
        }
        if (valorblue=='S') {
            enviaStatus(); // ##### Envia Status geral
            auTo();
        }
    } // Termina leitura Bluetooth
}

```

```

// ##### Leitura Botões

estadobotao=digitalRead(botao);
delay(50);
if (estadobotao==!HIGH){
  enviaStatus();
} // Botão Manual pressionado

// ##### Velocidade e a distância.
odometro= (((contOdometro*perimetro)/2)/1000);

if (millis() - lastRefresh >= 998) {
  detachInterrupt(digitalPinToInterrupt(2)); // Desativa a interrupção enquanto envia os dados
  vel =((3600*perimetro*contador)/2)/(millis() - lastRefresh);
  lastRefresh = millis();
  String velocidade="velocidade=" + String(vel)+"=";
  serialdoblueetooth.print(velocidade);
  String distancia="distancia=" + String(odometro)+"=";
  serialdoblueetooth.print(distancia);
  contador = 0;
}
  attachInterrupt(digitalPinToInterrupt(2), velocidadeCalc, RISING); // Ativa a interrupção novamente
  t.update(); // ##### Atualiza o time
}

```

APÊNDICE C – APLICATIVO APP INVENTOR 2

The image displays 15 screenshots of Scratch code blocks, organized into 15 numbered sections (1-15). The code is written in Portuguese and implements various functions for an application named 'APP INVENTOR 2'. The sections include:

- 1:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 2:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 3:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 4:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 5:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 6:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 7:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 8:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 9:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 10:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 11:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 12:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 13:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 14:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.
- 15:** A 'quando clicado' event block with a 'se' condition. If 'se' is true, it calls 'mostrar global' and 'esconder global'. If false, it calls 'mostrar global' and 'esconder global'.

APÊNDICE D – PESQUISA DE VALIDAÇÃO DO PROTÓTIPO

