



**DÉBORA DO AMARAL GOULART E SILVA**

**DESENVOLVIMENTO DE UM SIMULADOR  
BASEADO EM NORMAS TÉCNICAS E SUA  
INTEGRAÇÃO A UM APLICATIVO DE  
CONTROLE DE UMA CADEIRA DE RODAS  
INTELIGENTE**

**LAVRAS – MG**

**2023**



**DÉBORA DO AMARAL GOULART E SILVA**

**DESENVOLVIMENTO DE UM SIMULADOR BASEADO EM NORMAS  
TÉCNICAS E SUA INTEGRAÇÃO A UM APLICATIVO DE CONTROLE  
DE UMA CADEIRA DE RODAS INTELIGENTE**

Monografia apresentada à Universidade Federal de  
Lavras, como parte das exigências do Curso de  
Engenharia de Controle e Automação, para obtenção  
do título de Bacharel.

Prof. Dr. Danilo Alves de Lima  
Orientador

**LAVRAS – MG**

**2023**

**DÉBORA DO AMARAL GOULART E SILVA**

**DESENVOLVIMENTO DE UM SIMULADOR BASEADO EM NORMAS  
TÉCNICAS E SUA INTEGRAÇÃO A UM APLICATIVO DE CONTROLE  
DE UMA CADEIRA DE RODAS INTELIGENTE  
DEVELOPMENT OF A SIMULATOR BASED ON TECHNICAL  
STANDARDS AND ITS INTEGRATION TO AN INTELLIGENT  
WHEELCHAIR CONTROL APPLICATION**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Engenharia de Controle e Automação, para obtenção do título de Bacharel.

APROVADA em 24 de Fevereiro de 2023.

Arthur de Miranda Neto      UFLA  
Leomar dos Santos Marques   UFJF

Prof. Dr. Danilo Alves de Lima  
Orientador

**LAVRAS – MG  
2023**

*À minha família, por todo o incentivo, apoio e carinho.*



## **AGRADECIMENTOS**

Agradeço à Deus por me dar força e sabedoria. Aos professores da Universidade Federal de Lavras, pelos ensinamentos e por compartilharem o conhecimento que foi fundamental na realização deste trabalho. Agradeço o professor Dr. Danilo Alves de Lima, pela orientação e, principalmente, pela confiança na minha pesquisa, pela disponibilidade e pela paciência. Sou agradecida à minha família, pelo apoio incondicional e pela confiança em minha caminhada, ao meu namorado Rodolfo, pela compreensão, paciência, incentivo e apoio em todos os momentos. Gratifico, também, o professor Dr. Arthur de Miranda Neto e Leomar dos Santos Marques pelos conhecimentos repassados e por fazerem parte da minha trajetória acadêmica.



*Por vezes sentimos que aquilo que fazemos não é senão uma gota de água no mar. Mas o mar seria menor se lhe faltasse uma gota.  
(Madre Teresa de Calcuta)*



## RESUMO

Com os avanços atuais em acessibilidade, observa-se que o segmento de pesquisa relativo à Robótica Assistiva e sua aplicação em cadeiras de rodas tem ganhado evidência. Para uma melhor comunicação com o usuário, esses sistemas robóticos necessitam de interfaces homem-máquina e sistemas de auxílio à navegação. No entanto, sua concepção final, ainda de custo elevado, requer avanços para difundir a utilização em diferentes segmentos da sociedade. Com essa visão, o Laboratório de Mobilidade Terrestre (LMT) da Universidade Federal de Lavras (UFLA) iniciou o desenvolvimento de um protótipo de baixo custo para tornar cadeiras de rodas convencionais em motorizadas e inteligentes. Este trabalho tem como objetivo integrar uma interface projetada com suporte a comandos de voz utilizando um smartphone, a fim de realizar o controle do sistema motorizado com todas as características físicas em Ambiente Virtual (AV). Além disso, ele visa contribuir com investigações no processo de prototipagem desse sistema. Os testes em simuladores funcionam como um processo de treinamento, oferecendo suporte para que o usuário possa se adaptar às novas formas de condução de uma cadeira de rodas e também como uma maneira de avaliar as características funcionais do sistema, definindo uma metodologia de prototipagem rápida. Foram alcançados bons resultados na controlabilidade da cadeira de rodas no simulador pelo aplicativo, uma vez que, em todos os experimentos realizados, o operador concluiu o percurso proposto sem nenhuma colisão e com a correta identificação dos comandos direcionais. Para o futuro, poderão ser aplicadas técnicas de inspeção de usabilidade, incluindo voluntários cadeirantes nas etapas de teste, contribuindo dessa forma para o desenvolvimento de uma solução que possa atender as suas necessidades reais e cotidianas.

**Palavras-chave:** Ambiente Virtual. Smartphone. Cadeira de Rodas Inteligente.



## ABSTRACT

With the current advances in accessibility, it is observed that the research segment related to Assistive Robotics and its application in wheelchairs has gained evidence. For better communication with the user, these robotic systems need human-machine interfaces and navigation aid systems. However, its final design, still costly and requires advances to spread out its use in different segments of society. With this vision, the Terrestrial Mobility Laboratory (LMT) of the Federal University of Lavras (UFLA) started the development of a low-cost prototype to turn conventional wheelchairs into motorized and intelligent ones. This work aims to integrate an interface designed for voice commands using a smartphone, in order to perform control of the motorized system with all the physical characteristics in Virtual Environment (VA). In addition, it aims to contribute to investigations in the prototyping process of this system. The simulation tests work as a training process, offering support to user adaption to new ways of driving a wheelchair and also for safe evaluation of functional characteristics of the system, defining a fast prototyping approach. Good results were achieved in the controllability of the wheelchair in the simulator by the application, since in all the experiments carried out the operator completed the proposed route without any collision and with the correct identification of the directional commands. In future works, usability inspection techniques may be applied, including volunteers in wheelchairs in the test stages, contributing to the development of a solution that can meet your real and everyday needs.

**Keywords:** Virtual Environment. Smartphone. Smart Wheelchair.



## LISTA DE FIGURAS

Figura 2.1 – Equipamento acoplado em cadeira de rodas convencional simulado na plataforma Blender. . . . .	23
Figura 2.2 – Sistema adaptativo para cadeira de rodas em seu estágio atual.	24
Figura 2.3 – Exemplo do ambiente de trabalho no Unity3D. . . . .	25
Figura 2.4 – Tipos de sistemas de voz. . . . .	29
Figura 2.5 – Diagrama do modelo de reconhecimento de fala. . . . .	30
Figura 3.1 – Sensores e componentes presentes em smartphones. . . . .	32
Figura 4.1 – Fluxograma de interconexão dos dispositivos. . . . .	36
Figura 4.2 – Exemplo de Interface do Aplicativo. . . . .	37
Figura 4.3 – Exemplo de comunicação feita com o Google Now. . . . .	38
Figura 4.4 – Comando ‘frente 200’ detectado e em processamento. . . . .	38
Figura 4.5 – Fluxograma das etapas do sistema. . . . .	39
Figura 4.6 – Script C#. . . . .	41
Figura 4.7 – Inspector da cadeira de rodas com as variáveis do script. . . . .	42
Figura 4.8 – Espaço interno de uma cozinha desenvolvido no ambiente virtual. . . . .	44
Figura 4.9 – Rampa de acessibilidade para cadeira de rodas feita no ambiente virtual. . . . .	45
Figura 4.10 – Estacionamento com vaga paralela à calçada e rampa de acesso em ambiente virtual. . . . .	46
Figura 5.1 – Parâmetros ajustáveis da cadeira de rodas virtual. . . . .	49
Figura 5.2 – Ambiente virtual criado para a realização dos ensaios para validação dos freios do protótipo criado. . . . .	50
Figura 5.3 – Trajetória realizada em vermelho pela cadeira de rodas no Experimento 1. . . . .	52
Figura 5.4 – Trajetória realizada em vermelho pela cadeira de rodas no Experimento 2. . . . .	52

Figura 5.5 – Trajetória realizada em vermelho pela cadeira de rodas no Ex-  
perimento 3. . . . . 53

## LISTA DE TABELAS

Tabela 5.1 – Tempo de resposta para o comando “Frente 30”. . . . .	48
Tabela 5.2 – Resultados do ensaio de determinação de eficácia dos freios do protótipo proposto. . . . .	51



## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	19
1.1	Objetivos	21
1.2	Organização do Trabalho	22
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	23
2.1	Protótipo de sistema motorizado de baixo custo para cadeiras de rodas convencionais	23
2.2	Simuladores virtuais aplicados no desenvolvimento de projetos	24
2.3	Reconhecimento de Voz	27
<b>3</b>	<b>ESTADO DA ARTE</b>	31
3.1	Smartphones como dispositivos de controle	31
3.2	Ambientes Virtuais para cadeirantes	33
<b>4</b>	<b>METODOLOGIA</b>	35
4.1	O aplicativo	35
4.2	Comunicação	40
4.3	Controle	40
4.4	Ambiente Virtual	43
<b>5</b>	<b>RESULTADOS</b>	47
5.1	Validação do Simulador	49
5.2	Experimento 1	50
5.3	Experimento 2	51
5.4	Experimento 3	53
<b>6</b>	<b>CONCLUSÃO</b>	55
	<b>REFERÊNCIAS</b>	57



## 1 INTRODUÇÃO

Avanços tecnológicos têm permitido a inclusão de um maior grupo de usuários portadores de deficiências físicas nos meios de convívio social, mais especificamente com os progressos ocorridos nas estruturas das cadeiras de rodas. Dentre elas, destacam-se a melhoria dos materiais componentes (com metais de ligas leves), levando também à redução do peso, e ajustes ergonômicos que juntos garantem maior conforto e menor fadiga ao usuário.

As Cadeiras de Rodas Motorizadas (CRMs) surgiram como uma opção para demandar menos esforço físico dos cadeirantes, a maioria delas requer apenas a habilidade do usuário de controlar um *joystick* para indicar a direção desejada. Porém, existem pessoas que possuem deficiência tanto nos membros inferiores quanto superiores, tornando inviável esse tipo de controle.

A tetraplegia, por exemplo, geralmente é provocada por lesões que atingem a medula espinhal a nível da coluna cervical, ocasionando a perda dos movimentos dos braços, tronco e pernas. Essas lesões podem ser decorrentes de traumatismos em acidentes, hemorragia cerebral, sérias deformidades na coluna ou doenças neurológicas. A tetraplegia também recebe o nome de quadriplegia e se diferencia da paraplegia em razão do local da lesão medular. A paraplegia é decorrente de lesão medular abaixo da região torácica e afeta apenas o tronco e os membros inferiores, mantendo a força do braço (MINISTÉRIO DA SAÚDE, 2013).

Com a finalidade de contornar o problema da incapacidade de utilização do *joystick*, principalmente por pessoas tetraplégicas, surgem interfaces de controle alternativas que permitem guiar uma cadeira de rodas por outros meios. Existem estudos que descrevem técnicas, invasivas e não invasivas, utilizadas para fazer o controle dos sistemas, como pelo movimento do pescoço, por sopro e sucção e por Eletromiografia (EMG) (PINTO, 2016). Contudo, o estado da arte mostra que

a maioria desses projetos ainda possuem um alto custo de desenvolvimento, visto que os algoritmos e hardware utilizados são de alta complexidade.

Visando o conforto do usuário, sistemas com tecnologias não invasivas vêm ganhando espaço, a fim de garantir maior autonomia e liberdade. A utilização de comandos de voz é um exemplo disso. O cadeirante fica livre do contato com equipamentos e recursos externos que possam incomodá-lo. Já nos sistemas baseados em EMG, o usuário deve estar em contato com eletrodos enquanto estiver usando o dispositivo/cadeira de rodas, o que torna o sistema pesado e desconfortável.

Interfaces de controle podem incorporar recursos de comandos de voz. Quando essas soluções são direcionadas a dispositivos móveis, como *smartphones*, o acesso por parte das pessoas mais necessitadas torna-se mais fácil. Atualmente o *smartphone* se tornou uma ferramenta indispensável, muito mais do que um simples acessório, o *smartphone* é uma ferramenta cada vez mais essencial na vida, no trabalho, nos estudos e no lazer das pessoas.

No Brasil, o apoio às pessoas com deficiências motoras é pouco eficiente. Dados da Organização Mundial Da Saúde (OMS) (2012) sinalizam que a deficiência tem uma relação bidirecional com a pobreza, isto é, a deficiência pode aumentar o risco de pobreza e a pobreza aumenta o risco de deficiência. Cerca de 80% das Pessoas com Deficiência (PcD) vivem em países em desenvolvimento, segundo o Programa das Nações Unidas para o Desenvolvimento (PNUD). Há dados informando que ter alguma deficiência aumenta o custo de vida em cerca de 1/3 da renda e que mais de 50% das pessoas deficientes não têm condições de pagar por serviços de saúde.

Além de garantir o baixo custo, é imprescindível que uma tecnologia a ser desenvolvida seja segura e que o usuário final se sinta confortável ao utilizar o sistema. Conduzir uma cadeira de rodas sem o devido preparo, pode colocar em risco tanto o cadeirante como as pessoas à sua volta. Muitos são os desafios no manuseio

de uma cadeira de rodas, o percurso pode conter superfícies irregulares, rampas, ambientes estreitos e vários outros obstáculos que tornam a vida dos cadeirantes ainda mais difícil, evidenciando a necessidade de um ambiente de treinamento.

Com essa visão, iniciaram-se as pesquisas sobre as formas de controle e acionamento inteligente para realizar comandos de forma remota, direcionando o desenvolvimento da interface para o controle em Ambientes Virtuais (MARQUES et al., 2021a) e (MARQUES et al., 2021b). Neles, a aplicação se torna mais segura e o usuário é capaz de testar as principais consequências de cada ação sem risco no mundo real (RODRIGO; HERRERA, 2008). No simulador em forma de jogo, a pessoa é treinada para tomada de decisões, resolução de problemas e outros propósitos educacionais (HERRLICH et al., 2010). Além disso, como o Ambiente Virtual simula situações reais, ele possibilita que qualquer pessoa, com deficiência ou não, possa experimentar algumas dificuldades enfrentadas no cotidiano daquelas que, por alguma deficiência motora, precisam utilizar uma cadeira de rodas (CORRÊA et al., 2020).

Nesse sentido, o presente trabalho visa a integração de um aplicativo de controle de cadeira de rodas a um simulador baseado em normas técnicas, a fim de validar e treinar futuros usuários da aplicação. Utilizando um *smartphone* e o recurso de reconhecimento de voz presente no aplicativo, o celular do usuário se torna um motor de processamento.

## 1.1 Objetivos

Este trabalho tem por objetivo o desenvolvimento de Ambientes Virtuais baseados em normas técnicas e sua integração com uma aplicação de reconhecimento de voz para controle de cadeira de rodas, permitindo testes e validações futuras, dando prosseguimento aos trabalhos já desenvolvidos no Laboratório de Mobilidade Terrestre (LMT) nesta área. Para tanto, os seguintes objetivos específicos são contemplados:

- Estudar os requisitos do usuário em função da tarefa para definir qual a ferramenta é a mais adequada; quais são as técnicas já existentes e suas principais características;
- Desenvolver cenários em Ambiente Virtual exclusivo para simulação com cadeiras de rodas, dimensionado de acordo com a norma de acessibilidade NBR 9050:2004.
- Desenvolver aplicativo Android com suporte para reconhecimento de voz de forma contínua, sem necessidade de interação por toque e apto para troca de dados com o computador (Ambiente Virtual);
- Realizar ensaios de teste a fim de verificar se as etapas de fabricação do produto seguiram os requisitos exigidos pela Associação Brasileira de Normas Técnicas, especialmente na parte 3, na Determinação da Eficácia dos Freios;

## **1.2 Organização do Trabalho**

Este documento está organizado da seguinte maneira: o Capítulo 2 apresenta os conceitos teóricos que embasaram o desenvolvimento deste trabalho. O Capítulo 3 é composto por conteúdos de pesquisas acadêmicas relacionadas aos temas tratados aqui. O Capítulo 4 discute a definição da metodologia usada na aplicação, onde são feitas descrições dos procedimentos para o desenvolvimento do simulador baseado em normas técnicas e sua integração ao aplicativo. No Capítulo 5 são exibidos os resultados de cada experimento proposto. Finalmente, no Capítulo 6, as conclusões gerais deste trabalho são apresentadas. Paralelamente, as perspectivas de continuidade são indicadas, exibindo os possíveis caminhos a serem seguidos como seguimento deste trabalho.

## 2 REFERENCIAL TEÓRICO

Neste capítulo, serão abordados os conceitos teóricos que serviram de base para o desenvolvimento do presente trabalho, sendo eles: protótipo de sistema motorizado de baixo custo para cadeiras de rodas convencionais, simuladores virtuais aplicados no desenvolvimento de projetos e reconhecimento de Voz.

### 2.1 Protótipo de sistema motorizado de baixo custo para cadeiras de rodas convencionais

A elaboração deste trabalho tem como base um protótipo de baixo custo para tornar cadeiras de rodas convencionais em motorizadas e inteligentes que está sendo desenvolvido no Laboratório de Mobilidade Terrestre (LMT) na Universidade Federal de Lavras (UFLA). O projeto realiza análises de tensões e deformações utilizando o método dos elementos finitos, com a finalidade de avaliar os aspectos funcionais e estruturais por meio de simulações em uma cadeira de rodas convencional (MARQUES et al., 2019). Ele leva em consideração técnicas e fundamentos ergonômicos, antropométricos, biomecânicos e normativos vigentes no país, com o objetivo de garantir às pessoas com deficiência motora o direito de locomoção de forma autônoma e independente. A Figura 2.1 ilustra o sistema completo feito no ambiente de simulação gráfica Blender.

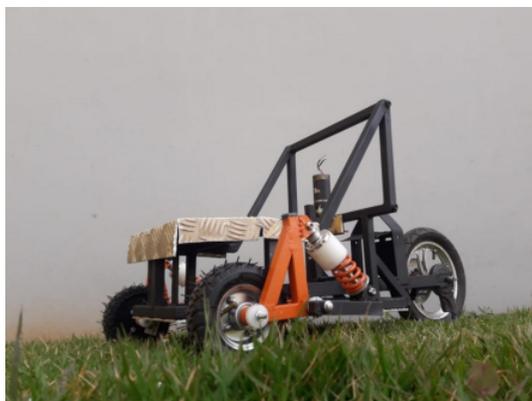
Figura 2.1 – Equipamento acoplado em cadeira de rodas convencional simulado na plataforma Blender.



Fonte: Arquivo pessoal

A definição da posição da estrutura de suporte a ser colocada na parte frontal e na área de apoio dos pés foi dada por meio de três estudos. O primeiro referente à análise do INMETRO, o qual reprovou todas as cadeiras no suporte de pé. O segundo, por sua vez, considera elementos finitos (MARQUES et al., 2019), onde também evidencia o suporte do pé com características que o reprova. Já o terceiro realiza um estudo de colisão de cadeiras de rodas, onde se propõe como solução estruturas de impacto para proteção do indivíduo. Na Figura 2.2 é ilustrado o estado atual do protótipo em desenvolvimento por (MARQUES et al., 2019) em parceria com o LMT.

Figura 2.2 – Sistema adaptativo para cadeira de rodas em seu estágio atual.



Fonte: Arquivo pessoal

## 2.2 Simuladores virtuais aplicados no desenvolvimento de projetos

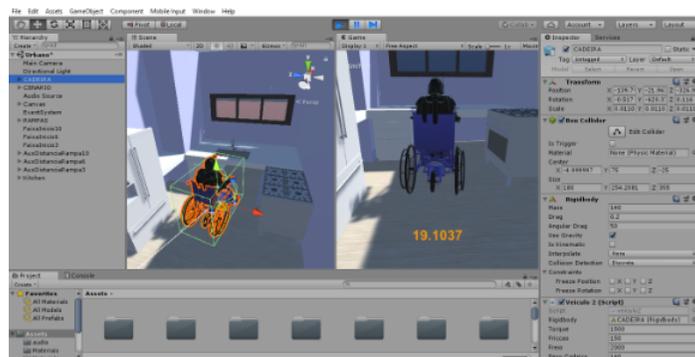
Ambientes Virtuais (AV) permitem o desenvolvimento de habilidades específicas e evoluem de forma independente da simulação com a finalidade de criar um ambiente de aprendizagem seguro, onde o usuário pode testar as principais consequências de cada ação sem risco no mundo real. Com uma simulação, é possível expandir o leque de situações possíveis e adaptá-las a diferentes usuários, capacidades e conhecimentos (RODRIGO; HERRERA, 2008). Os estudos para desenvolvimento de AV aplicados a projetos têm sido amplamente investiga-

dos em diversos âmbitos. Eles têm como finalidade facilitar tanto a construção de novas tecnologias quanto o treinamento de usuários em diversos tipos de interfaces de comando, fornecendo uma base útil para otimizar diversos dispositivos (COOPER et al., 2002). Dessa forma, reduzem os custos com o desenvolvimento de protótipos e contribuem na definição de novos requisitos para o projeto antes mesmo de sua fabricação.

O Unity é um motor gráfico de desenvolvimento multiplataforma muito utilizado para criação, desenvolvimento de jogos e experiências interativas em 2D e 3D. Por ser multiplataforma, permite o desenvolvimento de aplicações para iOS e Android por exemplo.

O desenvolvimento no motor de jogos é bem simples e se faz por meio do Editor Unity. Nele é possível organizar a interface que será apresentada ao usuário, composta de modelos gráficos e sonoros, que são ajustados pelo desenvolvedor a fim de se criar o visual do ambiente desejado. Um exemplo do ambiente de desenvolvimento é mostrado na Figura 2.3.

Figura 2.3 – Exemplo do ambiente de trabalho no Unity3D.



Fonte: Do autor.

Para estabelecer as regras e eventos do ambiente criado, são utilizados *scripts* na linguagem C#. A edição dos códigos é feita por meio do MonoDevelop ou Microsoft Visual Studio.

A ferramenta Unity também está presente em iniciativas muito interessantes, como em aplicativos de Realidade Mista, para treinamento de cadeira de rodas elétricas (DAY; JOHN, 2019). Em aplicações como essa, a cadeira de rodas não é processada dentro do ambiente, porém, para permitir que ela interaja com os objetos virtuais, um grupo de colisores físicos são usados para corresponder à forma genérica de uma cadeira de rodas. Os colisores são recursos muito explorados na ferramenta Unity. Eles são usados para detecção de colisão, podendo indicar o momento em que a cadeira de rodas colide com uma mobília virtual, por exemplo, nesse momento, a interface do usuário é atualizada e ele recebe uma indicação por ruído que se assemelha a algo atingindo um objeto.

O Unity 3D oferece várias funcionalidades, e uma de suas principais vantagens é a possibilidade de integrar dispositivos de entrada e saída. Além de que, o Unity oferece a Unity Asset Store. Essa loja contém vários elementos que podem ser facilmente adicionados aos games, como *scripts*, animações e efeitos, por exemplo. Assim, basta o desenvolvedor analisar a loja, verificar as opções e montar um jogo rico em detalhes visuais. Alguns deles, inclusive, são gratuitos (DEVMEDIA, 2023).

O Unity 3D também se destaca por ter uma comunidade ativa e com foco na colaboração, documentações extensas e manuais de usuário. Recursos esses que auxiliam os desenvolvedores mais inexperientes.

No entanto, o Unity não é a única opção disponível para o desenvolvimento de jogos que oferece diversos recursos e ferramentas. Há outras, como o Unreal Engine, que também pode ser usado com o mesmo propósito, inclusive em aplicações como os ambientes de treinamentos para cadeirantes (HERRLICH et al., 2010).

O Unreal é considerado um engine com boa qualidade de gráficos visuais e também é conhecido por ser usado em grandes estúdios. Porém, tanto o Unity

3D quanto o Unreal, são capazes de oferecerem as mesmas configurações e funcionalidades.

A decisão pelo uso do Unity 3D se deu pelo fato de ser considerada uma ferramenta um pouco mais intuitiva. O Unity tem seu uso mais difundido entre a comunidade de desenvolvedores, o que faz dele a principal escolha por aqueles que não contam com equipes extensas no processo de criação.

### **2.3 Reconhecimento de Voz**

As investigações na área do reconhecimento de fala foram iniciadas na década de 30 e marcadas principalmente pela criação do VOCODER, o primeiro sintetizador eletrônico de voz nos laboratórios Bell, pelo engenheiro Homer W. Dudley. O VOCODER (ou Voice Coder) tinha como propósito reduzir a largura de banda necessária para a transmissão de fala, de forma que um maior número de chamadas telefônicas pudessem ser transmitidas em uma determinada linha (GOMES, 2007). Nessa época, grande parte dos financiamentos iniciais das pesquisas eram feitos com base na motivação de se utilizar tecnologias para espionagem e decodificação de mensagens. Os primeiros resultados em relação ao reconhecimento de voz foram aparecendo ao longo da década de 40. Visto que, para uma evolução nas tecnologias de fala, o desenvolvimento de sistemas práticos e com maior qualidade, dependiam diretamente de computadores e conversores AD/DA (Analog-Digital/Digital-Analog), recursos que somente surgiram em meados da década de 60. No entanto, é a partir dos anos 1980, já com um avanço computacional e com o amadurecimento do reconhecimento de fala contínuo, que as tecnologias de reconhecimento de fala surgiram comercialmente (GOMES, 2007).

Os desafios computacionais no reconhecimento de fala continuaram existindo, visto que o sinal digital de fala é dinâmico e sempre apresenta uma estrutura diferente, mesmo quando pronunciado pelo mesmo locutor (ALVARENGA, 2012). Desafios como este, passaram a impulsionar pesquisadores no desenvol-

vimento de soluções que pudessem se aproximar de um cérebro humano. Assim como a poderosa ferramenta que é o cérebro humano, precisava-se de um recurso capaz de reconhecer padrões e identificar comandos de voz independente do locutor, visando ao aperfeiçoamento da interação homem-máquina.

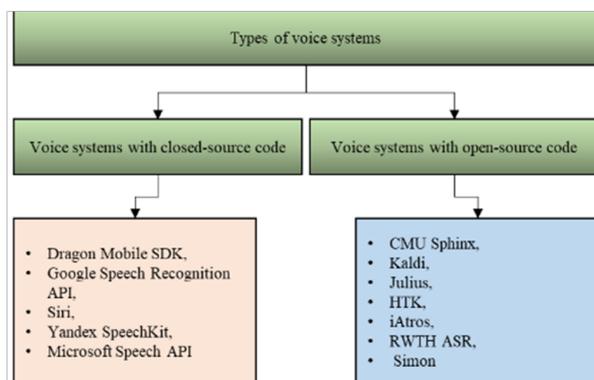
Técnicas como os Modelos Ocultos de Markov (HMM - *Hidden Markov Model*), Coeficientes Cepstrais de Frequência Mel (MFCC - *Mel-frequency Cepstral Coefficients*) e Redes Neurais Artificiais (ANN - *Artificial Neural Networks*), se tornaram conhecidas. A combinação delas, em diferentes fases do reconhecimento, é capaz de trazer melhoria significativa de desempenho (RABINER; JUANG, 1993). A partir disso, as pesquisas já passaram a contar com a utilização de tecnologia de redes neurais. Atualmente, sistemas de Reconhecimento Automático de Fala (ASR - *Automatic Speech Recognition*) e até detecções mais específicas, como o nível de estresse no sinal de fala podem ser implementados inteiramente por modelos baseados em Redes Neurais Profundas (DNN - *Deep Neural Network*) (SILVA; SEARA, 2020) e Aprendizado Profundo (*Deep Learning*) (HAN; BYUN; KANG, 2018).

Nos dias de hoje, com o avanço da Internet das Coisas, em inglês *Internet of Things* (IoT), observa-se cada vez mais a utilização do reconhecimento de voz em diversos tipos de aplicações, a fim de disponibilizar estímulos para ações automáticas de determinados equipamentos. Ligar e desligar as luzes de uma residência, acionar eletrodomésticos e ar condicionado e realizar ligações telefônicas são algumas das diversas possibilidades existentes utilizando um simples comando de voz.

Os sistemas de reconhecimento de fala podem ser divididos em dois grupos, sistemas com códigos-fonte fechado e sistemas com código-fonte aberto. Os tipos de sistemas são exemplificados na Figura 2.4.

Frente às possibilidades, pode se dizer que o sistema mais adequado para o uso em *smartphones* são os mecanismos com código fechado, uma vez que já

Figura 2.4 – Tipos de sistemas de voz.



(MATARNEH et al., 2017)

estão prontos para o uso e acabam se destacando por sua simplicidade, podendo o desenvolvedor usá-la de maneira fácil e rápida (MATARNEH et al., 2017).

Quando se trata do uso em dispositivos móveis, os recursos que mais se destacam são as Interfaces de Programação de Aplicativos, em inglês *Application Programming Interface* (API), devido a sua economia e eficiência. Existem muitas APIs abertas e prontas para uso pelos desenvolvedores, *SpeechRecognizer API*, *Google Cloud Speech* e *Microsoft Bing Speech* são alguns exemplos. Porém não existe uma única API com solução universal que se adeque a todos os projetos, cada uma delas tem suas características e nem todos os requisitos podem ser atendidos por um sistema. Por isso, é preciso selecionar uma delas para se preservar a consistência do projeto (DEVELOPEX, 2022).

Pela praticidade de utilização e por já estar integrada ao Android, optou-se pela *SpeechRecognizer API*. A implementação dessa API transmite áudio para servidores remotos para realizar o reconhecimento de fala (ANDROID DEVELOPER, 2022) e o resultado é apresentado como um arquivo *String*. Na Figura 2.5 é ilustrado o diagrama desse procedimento que se inicia com a gravação do áudio do locutor e termina na obtenção do resultado, que por sua vez é exibido na tela da interface.

Figura 2.5 – Diagrama do modelo de reconhecimento de fala.



Fonte: Do autor

Porém, é importante destacar que, assim como em outras aplicações, também existem dificuldades em se utilizar a API do Google. Apesar de muito eficiente na transcrição de fala, a API não é recomendada para o uso em produção por conter algumas limitações. O sistema funciona somente por sessões curtas, de aproximadamente 1 minuto, o que é suficiente para reconhecer palavras separadas ou até frases curtas, mas não discursos muito extensos. Além disso, a API não pode ser totalmente personalizada, fazendo com que nem todos os requisitos do desenvolvedor possam ser atendidos durante o desenvolvimento de sua aplicação (DEVELOPEX, 2022).

### 3 ESTADO DA ARTE

Tendo em vista as necessidades e tecnologias já citadas, existem muitas iniciativas para criar cadeiras inteligentes pelo mundo, utilizando várias interfaces e ideias diferentes. Neste capítulo, são descritas algumas das iniciativas que possuem maior sinergia com este trabalho e que de alguma forma contribuíram com os conceitos e ideias que são demonstrados.

#### 3.1 Smartphones como dispositivos de controle

Existem vários exemplos de aplicações de baixo custo que integram assistência ao usuário e *smartphones*, como em (SOARES; LIMA; NETO, 2019), onde foi desenvolvida uma aplicação móvel de baixo custo, responsável pelo monitoramento do condutor e pela identificação de sinais de sonolência e de adormecimento baseados em análises dos estados dos olhos. A aplicabilidade dos *smartphones* como um sensor inteligente para veículos inteligentes é destacada por Neto, Lima e Neto (2016), mostrando que esses dispositivos móveis são capazes de fornecer um alto poder computacional aliado a múltiplos sensores embarcados e capacidades de conexão sem fio. São, portanto, uma plataforma conveniente e prática para se usar como um sensor inteligente e assistência de direção para veículos inteligentes.

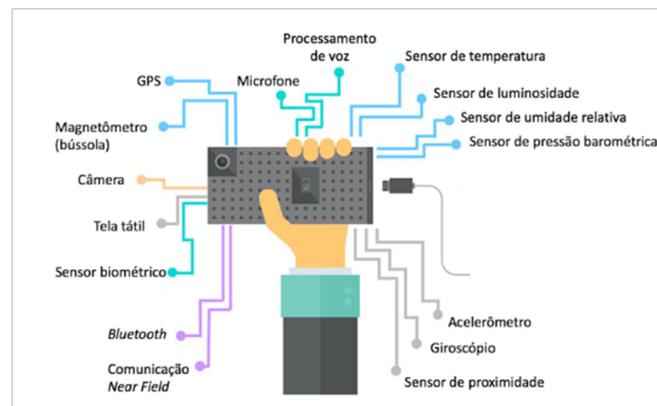
A proposta de Azevedo, Walcimar e Eduardo (2016) apresenta uma Interface Homem-Máquina (IHM) em plataforma Android, com baixo custo computacional e totalmente embarcada em dispositivo celular para controle de dirigibilidade de uma cadeira de rodas motorizada. A comunicação entre o aplicativo e o microcontrolador é feita por módulo Bluetooth, a interface de acionamento utiliza tanto comandos de voz como toque em tela e o processamento de voz é feito em nuvem com recursos do Google.

Em (NASCIMENTO, 2018), é mostrado o desenvolvimento de uma ferramenta para o planejamento de rotas com o objetivo de contribuir para a locomoção

dos usuários de cadeiras de rodas pelas vias urbanas. Um trabalho que exemplifica a utilização de diferentes recursos do Google (*Google Maps*) para soluções de problemas de mobilidade presentes no cotidiano de pessoas com deficiências físicas.

Dias et al. (2018) demonstram o desenvolvimento de uma interface para cadeirantes com deficiência motora severa para permitir que os mesmos interajam com equipamentos do ambiente em sua volta por meio do movimento da cabeça. A pesquisa é utilizada como base para o entendimento da aplicação de outros tipos de interface no controle de uma cadeira de rodas. Ela ainda ressalta o uso dos *smartphones* como um recurso poderoso no contexto de Tecnologias Assistivas (BYROM et al., 2016), selecionando o dispositivo como fonte de dados inerciais necessários para a inferência do ângulo da orientação da cabeça do usuário, para interação com o ambiente. Na Figura 3.1 estão alguns dos recursos presentes nos *smartphones*.

Figura 3.1 – Sensores e componentes presentes em smartphones.



(DIAS et al., 2018)

Analisando as informações presentes nos trabalhos selecionados, conclui-se que os *smartphones* são um recurso completo e de fácil acesso. Além disso, apresentam a possibilidade e facilidade de integração de novas funcionalidades, dependendo das necessidades e condições físicas do usuário. O diferencial do

trabalho desenvolvido se dá pelo controle totalmente remoto, sem a necessidade de toque em tela e por se tratar de uma forma de controle não invasiva, sendo confortável para o usuário

### **3.2 Ambientes Virtuais para cadeirantes**

Existem vários tipos de simuladores na literatura, destinados para fins de aprendizado de direção, teste de habilidades, desempenho de direção e testes de interfaces de entrada.

Em Pithon et al. (2009) foram fornecidas algumas evidências a respeito de simuladores de cadeira de rodas. A análise indica que os simuladores podem ser eficazes e aumentar a motivação dos usuários no uso da tecnologia desenvolvida.

Uyanik et al. (2022) demonstram o desenvolvimento de uma casa inteligente e um aplicativo de controle de cadeira de rodas em ambiente virtual. O simulador foi criado com o auxílio da ferramenta Unity 3D. O ambiente virtual possui vários tipos de eletrodomésticos da vida cotidiana e uma cadeira de rodas, que é controlada pelo usuário com base no sistema de interface cérebro-computador (BCI).

O trabalho de Corrêa et al. (2020) foca-se no desenvolvimento de um ambiente 3D para simulações de situações reais, tornando possível que qualquer pessoa, com deficiência ou não, consiga ter a sensação de utilizar uma cadeira de rodas elétrica e vivenciar algumas das muitas dificuldades enfrentadas no dia-a-dia dos cadeirantes. O autor seleciona e propõe 3 cenários distintos, sendo eles a parte externa de uma residência, um estacionamento e a sala de estar.

Ambientes Virtuais sejam eles internos ou externos, imersivos ou não, estão disponíveis na literatura. Alguns são criados para testar a percepção do usuário, outros para avaliar as características físicas de uma cadeira de rodas, ou até mesmo destinados a serem jogos educativos. Contudo, não se tem conhecimento de Ambientes Virtuais projetados para seguirem as recomendações de dimensionamento

estipulados por normas técnicas a fim de validar dispositivos inteligentes. O presente trabalho foca na aplicação dessas normas de acessibilidades para cadeirantes, seguindo os requisitos exigidos, a fim de garantir qualidade e segurança.

## 4 METODOLOGIA

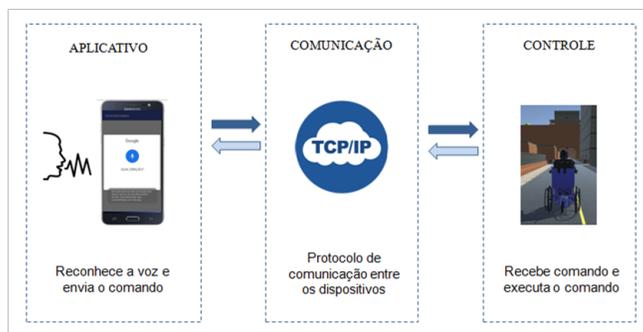
O controle inteligente da cadeira de rodas foi dividido em três etapas de desenvolvimento. A primeira correspondeu à criação da IHM, sendo o layout definido para suportar os comandos de voz. Na segunda etapa foi definida a comunicação via Sockets entre *smartphone* e o computador que executa o simulador, utilizando o protocolo TCP/IP de maneira que um fluxo de bytes (mensagem de comando) pudesse ser transportado pela rede de forma rápida e segura. A terceira etapa foi a programação do servidor para fazer o controle da simulação. Nela estabeleceu-se um tipo de controle em malha aberta e, a partir dos comandos recebidos da rede com o protocolo TCP, as ações com a direção desejada foram sendo realizadas. Ou seja, as informações recebidas pelo servidor, como o ângulo de giro, são usados para movimentar direcionalmente a cadeira de rodas no Ambiente Virtual.

Para a realização dos testes da interface de controle da cadeira de rodas via simulação, utilizou-se o Unity 3D, ambiente onde foi desenvolvido o Simulador e onde foram desenvolvidos os códigos responsáveis pelo recebimento das informações de controle. Também foi utilizada a IDE do Android Studio, para desenvolver para a plataforma Android. Todos os testes da interface foram realizados utilizando um *smartphone* Samsung Galaxy S20 FE. Os códigos utilizados para a realização deste trabalho estão disponíveis no github (SILVA, 2023b). O fluxograma demonstrando o funcionamento geral do sistema, é mostrado na Figura 4.1.

### 4.1 O aplicativo

O aplicativo foi desenvolvido no software Android Studio, com suporte para reconhecimento de voz. O aplicativo funciona utilizando o microfone do *smartphone* para coleta do sinal de entrada. Este aplicativo possui a função de

Figura 4.1 – Fluxograma de interconexão dos dispositivos.



Fonte: Do autor

transformar em texto qualquer comando feito pelo usuário por meio da fala, usando a API *SpeechRecognizer* já integrada ao Android.

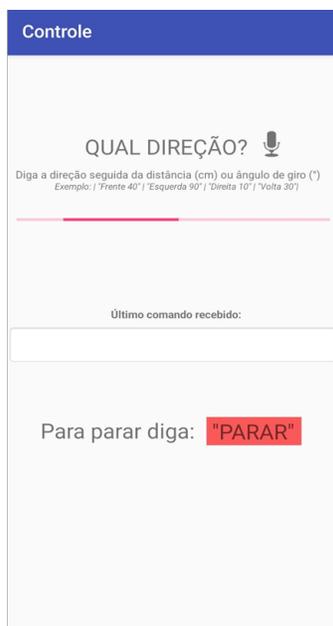
Optou-se por integrar ao aplicativo o serviço de reconhecimento de voz da Google. Por meio do *RecognizerIntent*, incorporado ao Android, é criada uma intenção de reconhecimento que solicita ao usuário um comando de voz iniciado com uma tela indicando os possíveis comandos para movimentação da cadeira, como mostrado na Figura 4.2.

Para iniciar o aplicativo no *smartphone* também de forma remota, foi acrescentada uma permissão capaz de integrar o Google Now, um assistente pessoal inteligente, disponível para o sistema operacional Android, com a aplicação desenvolvida no Android Studio.

Após o usuário ditar um comando, o áudio captado é enviado para reconhecimento e toda a parte de processamento do sinal é feita pelo serviço de reconhecimento de fala do Google, que retorna a palavra identificada por ele. Para realizar um controle suave e seguro da cadeira de rodas, o código foi dividido em tarefas (*threads*), permitindo executá-las em paralelo. Dessa forma, o sistema fica preparado para executar qualquer atividade emergencial que o usuário solicitar, mesmo quando algum outro comando já estiver sendo realizado.

O aplicativo possui suporte para o uso de comandos de voz, sem a necessidade de qualquer tipo de toque do usuário na tela do aparelho. Para iniciá-lo, basta

Figura 4.2 – Exemplo de Interface do Aplicativo.

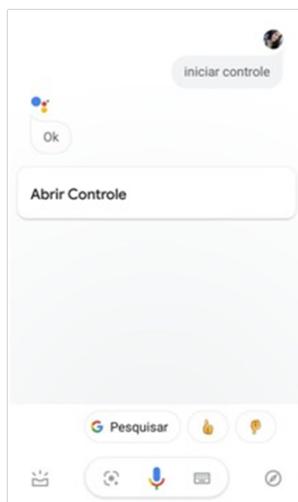


Fonte: Do autor

dizer “Ok, Google” seguido da ação com o nome da aplicação que se deseja abrir, no caso, “iniciar controle”. A tela mostrada na Figura 4.3 aparecerá e a interface do aplicativo da Figura 4.4 será exibida logo em seguida. Para a cadeira seguir em frente a uma distância de 200 centímetros, basta que o usuário pronuncie a palavra “frente”, seguida da quantidade de centímetros que deseja se locomover, nesse caso, “200”. O comando “frente 200” será processado e armazenado em uma matriz de *strings*. O resultado é enviado por meio do protocolo de comunicação entre os dispositivos e o usuário é informado na tela sobre qual o comando ele acabou de solicitar, assim como mostrado na Figura 4.4. O fluxograma da Figura 4.5 exemplifica de forma mais simplificada a sequência das ações.

Entradas de comandos e palavras consideradas inválidas não mudam o funcionamento do aplicativo, e também não são capazes de realizar movimentos de controle. Com isso, se o usuário se confundir e pronunciar um comando que não

Figura 4.3 – Exemplo de comunicação feita com o Google Now.



Fonte: Do autor

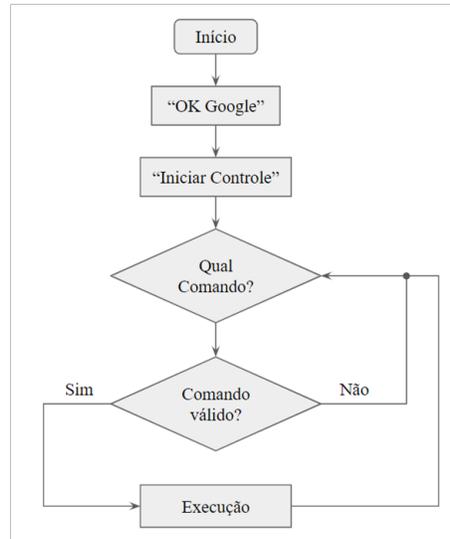
Figura 4.4 – Comando 'frente 200' detectado e em processamento.



Fonte: Do autor.

era o que realmente gostaria, a cadeira não realizará movimentos, apesar de que, toda e qualquer entrada dada por meio da fala será registrada na tela do aplicativo.

Figura 4.5 – Fluxograma das etapas do sistema.



Fonte: Do autor

O comando “frente 200” é enviado como uma *string*, em um formato específico, separada pelo símbolo de dois pontos. Nesse caso, a estrutura da *string* enviada é “frente:200”. A leitura desse comando é realizada no código do servidor por uma lógica que divide a *string* em duas partes, ou seja, uma variável recebe a informação da direção, e a outra recebe a distância ou o ângulo de giro.

Os comandos de direção aceitos pelo sistema são: ‘frente’, ‘direita’, ‘esquerda’, ‘parar’ ou ‘volta’. Para comandos como ‘direita’ ou ‘esquerda’ deve-se incluir o valor da angulação em graus e para comandos como ‘frente’, já mencionado anteriormente, é preciso especificar a quantidade em centímetros. Comandos ‘parar’ e ‘volta’ não necessitam de complemento.

Após qualquer um desses comandos, a cadeira de rodas na simulação então irá se locomover de acordo com o solicitado e o sistema ficará a espera de uma nova solicitação de movimento. Isso é possível visto que o aplicativo fica constantemente aguardando um novo comando.

## 4.2 Comunicação

A comunicação foi definida de forma que todos os dados enviados pelo aplicativo sejam coletados e processados no Ambiente Virtual, gerando então ações de controle. Essas ações são tratadas no módulo de controle, levando a realização das ações mecânicas, inicialmente fornecidas pelo usuário, fazendo dessa forma com que o sistema todo funcione de uma maneira dinâmica e amigável.

Em Ambiente Virtual, a comunicação acontece via Sockets, para isso foi criado um servidor, capaz de aceitar um cliente Android. Esse servidor é responsável por criar um socket (porta) que aceita a conexão do cliente. O cliente Android, por sua vez, cria um socket TCP local especificando o endereço IP do dispositivo onde o servidor foi iniciado e o número da porta do processo no servidor, com isso, a comunicação entre eles é estabelecida e sempre que o servidor estiver em execução o cliente poderá solicitá-lo para a troca de informações.

A perda de comunicação entre o aplicativo e o simulador pode ocorrer em casos em que a conexão Wi-Fi é perdida. Dependendo da situação, a interrupção pode acontecer no lado do simulador ou no lado do aplicativo, ou ainda nos dois ao mesmo tempo.

Caso a conexão Wi-Fi seja perdida somente do lado do computador, onde é executado o simulador, não há nada que indique visualmente que o controlador não está mais recebendo comandos. No entanto, ao se perder a conexão diretamente no aplicativo, as reações na tela da interface irão indicar ao usuário um erro de rede, informando-o que a partir desse momento os comandos de voz não estão sendo mais reconhecidos.

## 4.3 Controle

No Unity 3D cada objeto no jogo é um *GameObject* e para manipulá-los de forma a realizarem as ações desejadas é preciso que ele esteja associado a

um *script*. Os *scripts* na linguagem C# no Unity são inicialmente criados com a estrutura mostrada na Figura 4.6.

Figura 4.6 – Script C#.

```
1 01 using UnityEngine;
2 02 using System.Collections;
3 03
4 04 public class ScriptPlayer : MonoBehaviour
5 05 {
6 06     // Use this for initialization
7 07     void Start()
8 08     {
9 09
10 10     }
11 11
12 12     // Update is called once per frame
13 13     void Update()
14 14     {
15 15
16 16     }
17 17 }
```

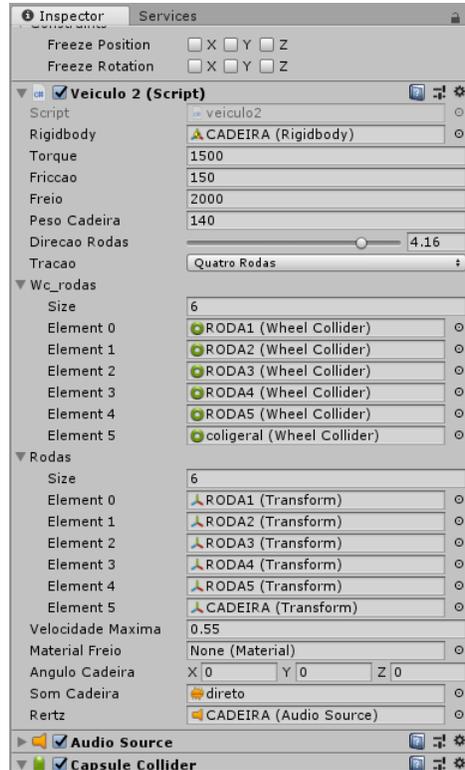
Fonte: Do autor

O método Start é chamado somente uma vez, na inicialização da cena, já o método Update é chamado a cada *frame*, de acordo com o clock/ciclo de máquina. Portanto, é por meio do código *script* dentro deste método que teremos os objetos do jogo se movimentando e alterando seus comportamentos.

Após a cadeira de rodas ser inserida no ambiente virtual, um *script* com a lógica para o controle foi associado a ela. O Unity possui uma opção para visualizar os componentes de cada objeto por meio da aba Inspector. O Inspector tem a finalidade de mostrar informações detalhadas sobre o *GameObject* selecionado no momento, incluindo todos os *Components* anexados a ele e suas propriedades. Na Figura 4.7 é possível verificar algumas das variáveis do *script* definidas para o controle.

As variáveis indicadas no Inspector podem ser alteradas em tempo de execução sem modificar o próprio *script*. Esse foi um dos recursos utilizados para a avaliação do simulador durante as etapas de validação e experimentos.

Figura 4.7 – Inspector da cadeira de rodas com as variáveis do script.



Fonte: Do autor

Um *script* base foi utilizado como ponto de partida para a controlabilidade da cadeira de rodas. Nesse *script* inicial, o controle da cadeira era feito utilizando o próprio teclado como dispositivo de entrada. O Unity possui uma class chamada Input, que contém métodos direcionados para cada tipo de dispositivo de entrada, inclusive o teclado, essa classe foi inicialmente usada como referência.

A partir disso, foi preciso adicionar funcionalidades no código, fazendo com que o simulador fosse capaz de receber e tratar as informações vindas do aplicativo. Dessa forma, iniciou-se a implementação do código em C# para criar o servidor e a lógica de recebimento e tratamento das mensagens de comando. Toda a lógica de controle foi feita em um *script*.

Como primeiro passo, cria-se uma thread, que é iniciada no método Start. Dentro dela o servidor é iniciado, por meio da classe TcpListener. Essa classe fornece métodos que escutam e aceitam solicitações de conexão de entrada. Também é nessa parte do código que informações como a porta de comunicação cliente/servidor e o endereço IP são definidos.

Como a estrutura da string de dados recebida pelo servidor é "frente:30", por exemplo, foi necessário fazer a leitura desse dado de forma a identificar o caractere ":" e armazenar as informações "frente" e "30" em variáveis separadas.

Após isso, é feita a verificação das variáveis isoladas por meio de estruturas condicionais e também realizadas as conversões necessárias dos tipos das variáveis.

No Unity 3D, todos os objetos do cenário são posicionados tridimensionalmente, assim, utilizando a propriedade Vector3 é realizada a manipulação das posições e direções 3D. Para rotacionar os GameObjects, o component Transform foi utilizado. Com isso, foi atribuído diretamente o valor recebido do aplicativo ao component. Fazendo com que fosse possível gerar reações de movimento na cadeira de rodas virtual.

Os testes de validação do simulador são baseados na Associação Brasileira de Normas Técnicas - ABNT (2015) mais especificamente na Parte 3 a respeito da determinação da eficácia dos freios. Considerando freios de serviço, operação normal e com ensaio aplicado para teste de freios de cadeiras motorizadas. Considerando também rampas fixas separadas. Os parâmetros considerados foram o torque, o atrito, a frenagem, o peso e velocidade máxima de 4.16 m/s, aproximadamente 15 km/h.

#### **4.4 Ambiente Virtual**

Os cenários utilizados no Ambiente Virtual foram previamente desenvolvidos na plataforma *SketchUp Online*, uma plataforma de modelagem 3D multi-

funcional, gratuita e de fácil usabilidade. Após a finalização do projeto na plataforma, foi feito o *download* do arquivo localmente e importado no Unity 3D, compondo todo o conjunto do simulador, fazendo a integração dos cenários com a cadeira de rodas.

Os ambientes projetados seguiram as recomendações de dimensionamento estipulados pela Norma Brasileira 9050/2004, denominada Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos, que tem como objetivo estabelecer parâmetros técnicos de projeto, construção, instalação e adaptação, visando o desenho universal (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT, 2004).

Segundo a Norma Brasileira 9050/2004, para a acessibilidade de pessoas em cadeira de rodas em cozinhas de dimensões mínimas, essas devem garantir uma área de aproximação frontal à pia, alcance manual confortável, com altura entre 0,80 m e 1,20 m, pia com altura máxima de 0,85 m e vão inferior livre de 0,73 m. Na Figura 4.8 é apresentado o cenário desenvolvido na plataforma *SketchUp Online*.

Figura 4.8 – Espaço interno de uma cozinha desenvolvido no ambiente virtual.



Fonte: Do autor

As rampas e rotas acessíveis não devem possuir ressaltos ou degraus em seu trajeto. A largura das rampas deve ser determinada pelo volume do fluxo de pessoas, recomendando-se largura mínima livre de 1,50 m, sendo a largura mínima admissível de 1,20 m. Deve-se instalar piso tátil de alerta, com faixas de largura

entre 0,25 e 0,60 m, distando-se até 0,32 m antes do início e após o final da rampa, para orientação de pessoas com deficiência visual. Devem existir patamares no início, no final e em cada segmento de rampa, com comprimento recomendado de 1,50 m, sendo a largura mínima admissível de 1,20 m (ASSOCIAÇÃO BRASILEIRA DE NORMAS TECNICAS - ABNT, 2004). O cenário desenvolvido a partir dessas especificações é mostrado na Figura 4.9.

Figura 4.9 – Rampa de acessibilidade para cadeira de rodas feita no ambiente virtual.



Fonte: Do autor

A Norma Brasileira 9050/2004 especifica que as vagas em estacionamento devem possuir largura mínima de 2,50m. Deve existir uma área livre de circulação com largura mínima de 1,20m, devidamente sinalizada horizontalmente, por meio de faixas inclinadas (“zebras”). Deverá existir uma rampa com acesso na calçada, com guia rebaixada, junto à área livre de circulação. As vagas também deverão possuir sinalização horizontal e vertical, utilizando o Símbolo Internacional de Acesso (SIA). A representação na plataforma *SketchUp Online* para a vaga de estacionamento pode ser vista na Figura 4.10.

O deslocamento em lugares estreitos, como em cozinhas, rampas e estacionamentos, que exigem movimentos limitados com curvas em 90 graus, demonstram a dificuldade que a maioria dos cadeirantes tem no dia a dia. Isso, mesmo acessando um local construído com respeito aos padrões estabelecidos.

Figura 4.10 – Estacionamento com vaga paralela à calçada e rampa de acesso em ambiente virtual.



Fonte: Do autor

Os testes realizados nesses ambientes foram utilizados para adaptar a IHM e o comportamento físico (características de direção) da cadeira de rodas e para detectar eventos importantes, como colisões entre a cadeira de rodas e objetos no mundo virtual.

## 5 RESULTADOS

O aplicativo de controle e o ambiente virtual foram desenvolvidos visando garantir qualidade e segurança para seus usuários. Porém, por mais que o sistema seja preciso, ainda irão existir falhas e alguns tipos de atraso no reconhecimento dos comandos de voz.

Uma das falhas que pode acontecer é a perda da comunicação entre o aplicativo e o simulador, isso ocorre quando a conexão com a internet é interrompida. Fazendo com que os dados não sejam trocados entre as aplicações. Uma demonstração ilustrando esse tipo de erro foi realizada e está disponível em vídeo (SILVA, 2023a). A conexão foi interrompida somente do lado do aplicativo, apenas para demonstrar as reações na tela da interface.

Foi observado que no momento em que a conexão Wi-Fi é perdida, surge uma mensagem de aviso no canto inferior, indicando que o aplicativo identificou um erro na rede. A partir desse momento, os comandos pronunciados não são mais processados pela interface. Porém, assim que a conexão é restabelecida, o aplicativo rapidamente volta ao seu funcionamento normal e o simulador consegue executar perfeitamente os dois últimos comandos ditados, “parar” e “direita 90”.

Caso a conexão Wi-Fi seja perdida somente do lado do computador, onde é executado o simulador, não há nada que indique visualmente que o controlador não está mais recebendo comandos. Nesse caso, quando se restabelece a conexão, a cadeira de rodas na simulação executa imediatamente e em sequência todos os comandos registrados no aplicativo durante o período em que estava sem conexão.

Em relação ao tempo de resposta do sistema, a análise foi realizada em 2 intervalos de tempo. O primeiro durante o processamento no aplicativo, entre o início do comando e o reconhecimento pelo aplicativo, e o segundo intervalo medido entre o instante de recebimento do comando pelo simulador e o início da movimentação da cadeira de rodas. Os resultados são apresentados na Tabela 5.1.

Tabela 5.1 – Tempo de resposta para o comando “Frente 30”.

<b>Teste</b>	<b>Tempo no Aplicativo (ms)</b>	<b>Tempo no Simulador (ms)</b>
1	2093	43
2	2097	35
3	2083	60
4	2068	73
5	2072	78
6	3025	34
7	2053	51
8	2058	69
9	2090	79
10	2058	52
<b>Média</b>	2169,7	57,4

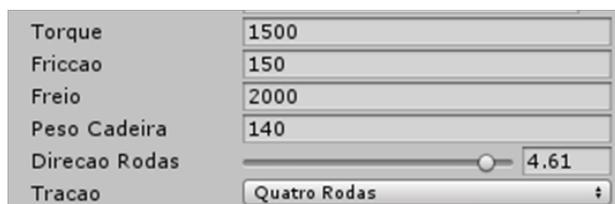
Fonte: do autor

Como nem todos os atrasos são possíveis de serem eliminados, optou-se por utilizar uma barra de progresso na interface, pela movimentação dessa barra é possível ter uma noção do momento em que o sistema está detectando a voz e quando ele finaliza a tarefa. Essa foi uma alternativa encontrada a fim de que o usuário saiba que em algumas situações ele deve esperar o tempo necessário para o sistema realizar suas ações.

Apesar do sistema apresentar atrasos no tempo de resposta, não foram identificadas situações em que esses atrasos prejudicassem a controlabilidade da cadeira de rodas durante a execução dos testes propostos.

Além disso, para explorar ainda mais o Ambiente Virtual, foram realizados ensaios de teste com a cadeira de rodas, a fim de verificar se as etapas de fabricação do produto seguiram os requisitos exigidos pela Associação Brasileira de Normas Técnicas (ABNT). Esses testes foram realizados apenas para validar e poder ajustar parâmetros, como a frenagem principalmente. Tração, fricção e peso também puderam ser analisados. Os parâmetros iniciais considerados na realização dos testes são indicados na Figura 5.1

Figura 5.1 – Parâmetros ajustáveis da cadeira de rodas virtual.



Fonte: Do autor

## 5.1 Validação do Simulador

Um dos ensaios realizados foi baseado na Associação Brasileira de Normas Técnicas - ABNT (2015), especialmente na parte 3, que trata da determinação da eficácia dos freios. Considerando que a velocidade máxima de uma cadeira de rodas motorizada exigida pela norma é de aproximadamente 4,16 m/s e que a distância de frenagem em operação normal não seja superior a 7 metros.

Para realizar esse teste, foi utilizado um ambiente externo, onde foram posicionadas as rampas de inclinação de 3, 6 e 10 graus. Após isso, foi criado um *script* adicional para o cálculo da velocidade, que foi associado ao *GameObject* da cadeira de rodas. O valor da velocidade foi escrito no console para facilitar a visualização. Para obter a distância percorrida após a frenagem, também foi necessária a criação de novos *scripts*, que fazem a comparação da distância entre dois objetos. Como referência para a medição dessas distâncias, foi utilizada a 'Roda5' da cadeira virtual, visto que é a mais distante da parte frontal da cadeira, ficando no meio central.

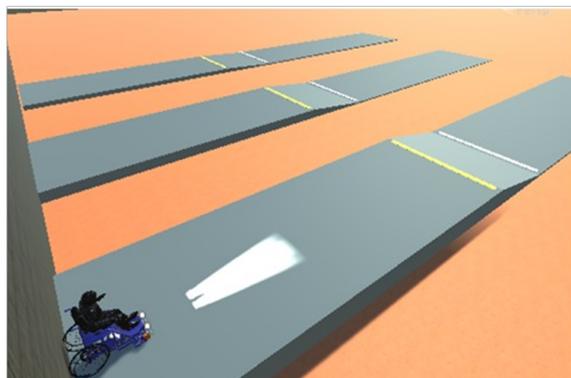
Com isso, iniciou-se os testes. Em plano horizontal, foi dado o comando para cadeira seguir em frente e ao atingir sua velocidade máxima acionou-se os freios. A distância indicada entre o início da operação dos freios e a cadeira de rodas finalmente chegando ao repouso foi medida.

O mesmo procedimento foi realizado nas rampas de inclinação de 3, 6 e 10 graus. Nessas rampas, foram adicionadas faixas com marcações que indicam

o início e final da inclinação das respectivas rampas. Essas marcações são usadas como referência para se obter a distancia percorrida após o acionamento dos freios.

Verificou-se com atenção qualquer comportamento anormal que pudesse surgir no momento do ensaio, como tombamento, deslizamento, falha do freio e desvio, por exemplo. Ao final, concluiu-se que com os parâmetros considerados a cadeira de rodas se encontra em conformidade com a norma no que diz respeito à eficácia dos freios. O ambiente desenvolvido para a realização dos testes é mostrado na Figura 5.2 e os valores de velocidade e de distância de frenagem obtidos nos ensaios podem ser verificados na Tabela 5.2.

Figura 5.2 – Ambiente virtual criado para a realização dos ensaios para validação dos freios do protótipo criado.



Fonte: Do autor

## 5.2 Experimento 1

Visando uma análise mais aprofundada do sistema, como já mencionado, desenvolveu-se também três cenários em Ambiente Virtual exclusivos para simulações com cadeiras de rodas, dimensionado de acordo com a norma de acessibilidade (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT, 2004). O primeiro ambiente escolhido tem como foco a mobilidade em cozinhas e copas.

Os testes realizados nesse tipo de ambiente apresentaram algumas dificuldades, já que, por ser um espaço pequeno são necessárias várias manobras para se

Tabela 5.2 – Resultados do ensaio de determinação de eficácia dos freios do protótipo proposto.

<b>Inclinação do plano de ensaio</b>	<b>Direção do movimento</b>	<b>Velocidade máxima (m/s)</b>	<b>Operação normal (m)</b>
Horizontal	De frente para	4,07	5,90
Horizontal	Reverso	4,05	6,86
3°	Voltada para a Descida	4,12	6,35
3°	De costas para a descida	4,10	6,40
6°	Voltada para a Descida	4,08	6,06
6°	De costas para a descida	4,02	6,74
10°	Voltada para a Descida	4,10	6,07
10°	De costas para a descida	4,11	6,74

Fonte: do autor

deslocar sem se colidir com os objetos ao redor. Em um teste básico, simulando a entrada da cadeira pela cozinha, até uma ida à pia e depois a volta até a geladeira e por fim o balcão, gastou-se um tempo de 1 minuto e 02 segundos e nas primeiras tentativas a cadeira acabou colidindo levemente com alguns objetos nos momentos de giro e ajustes de direção. A dificuldade na realização de manobras, em ambientes complexos como esse, pode ser solucionada por meio da repetição da trajetória. Quanto mais familiarizado o operador estiver com o ambiente a sua volta, menor será o tempo gasto nos deslocamentos e menores serão as chances de colisão durante a realização das manobras.

Foram pronunciados aproximadamente 8 comandos para a realização completa da tarefa proposta. A trajetória realizada pela cadeira pode ser vista na Figura 5.3. Um vídeo de demonstração mostra o experimento 1 sendo executado em (SILVA, 2023c).

### 5.3 Experimento 2

Os outros dois cenários desenvolvidos focam na mobilidade de cadeirantes em ambientes externos. O segundo ambiente é uma rampa de acessibilidade com curvatura de 90 graus, como pode ser visto na Figura 5.4. Nesse ambiente

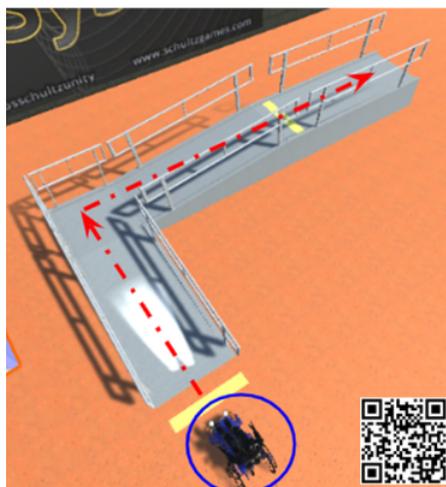
Figura 5.3 – Trajetória realizada em vermelho pela cadeira de rodas no Experimento 1.



Fonte: Do autor

foram realizados testes com o objetivo de comprovar a dirigibilidade da cadeira de rodas percorrendo toda a rampa sem colisões com as laterais. O resultado obtido foi satisfatório, visto que, com apenas 5 comandos e zero colisões, a cadeira completou o percurso no tempo de aproximadamente 45 segundos. Detalhes podem ser observados no vídeo disponível em (SILVA, 2023d).

Figura 5.4 – Trajetória realizada em vermelho pela cadeira de rodas no Experimento 2.



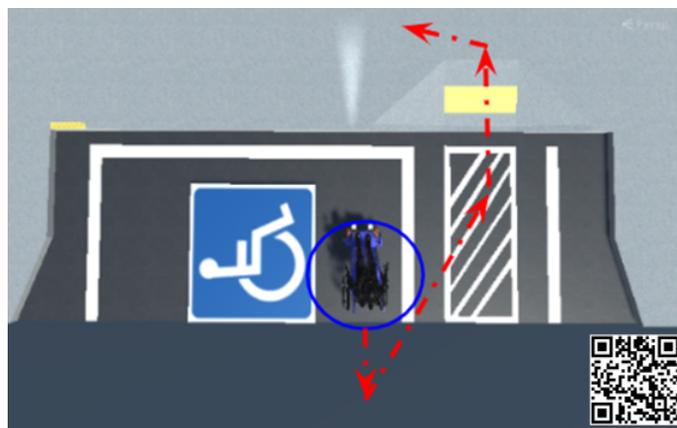
Fonte: Do autor

### 5.4 Experimento 3

O terceiro e último cenário, apresentado na Figura 5.5, corresponde a uma vaga de estacionamento paralela à calçada. Para finalizar o trajeto de sair da vaga e chegar até a calçada, gastou-se um tempo de 1 minuto e 01 segundo utilizando 7 comandos. O trajeto foi realizado de maneira simples e fácil, e por não haver objetos próximos para colisão o resultado da dirigibilidade nesse tipo de cenário pode ser considerado bem satisfatório. Detalhes da execução do experimento são apresentados no vídeo de demonstração em (SILVA, 2023e).

Vale ressaltar que apesar de ter obtido resultados satisfatórios nos experimentos realizados, é preciso considerar que em um sistema físico real existem ruídos. Esses ruídos fazem com que movimentos para frente, por exemplo, nem sempre sejam perfeitamente para frente. O simulador utilizado neste trabalho, busca representar o comportamento da cadeira de rodas o mais próximo da realidade possível, porém o controlador ainda não está configurado para considerar erros de odometria durante a execução dos comandos.

Figura 5.5 – Trajetória realizada em vermelho pela cadeira de rodas no Experimento 3.



Fonte: Do autor



## 6 CONCLUSÃO

Este trabalho teve por finalidade a integração de um aplicativo de controle de cadeira de rodas a um simulador, considerando-se para isso normas técnicas, a fim de validar e treinar futuros usuários da aplicação. Utilizando o próprio celular do usuário como motor de processamento, atrelado ao fato de se ter um sistema motorizado que pode ser facilmente adaptável a qualquer cadeira de rodas manual, é garantida ampla aplicabilidade.

Uma das dificuldades encontradas durante o desenvolvimento foi no deslocamento em ambientes estreitos, como em cozinhas, por exemplo. Apesar do operador concluir os experimentos propostos, em alguns casos, são necessárias várias manobras e muitos comandos para pequenos deslocamentos. Em ambientes reais, isso pode gerar fadiga no usuário e até abandono do sistema por falta de adaptação. Como forma de contornar esse problema, pode-se explorar ainda mais o simulador, utilizando-o como ambiente de treinamento, de forma que, por meio da repetição da trajetória, o operador possa identificar a melhor forma de realizar manobras, podendo reduzir o tempo gasto na realização de trajetórias em ambientes complexos. Outra alternativa seria a utilização de sensores para aprimorar o controle com base na resposta do sistema.

Também é possível combinar o controle por voz com outras modalidades, como o controle por expressões faciais e movimentos dos olhos, por exemplo, criando assim interfaces multimodais. Deste modo é possível adaptar o sistema conforme o grau de severidade da deficiência do usuário.

Vale ressaltar que o sistema foi projetado em malha aberta, existindo o risco associado a perda de comunicação. A execução de comandos não identificados durante as falhas de conexão geram preocupação e podem ser contornadas adicionando-se um processo de verificação do último estado da cadeira antes da execução da lista de comandos não executados.

Nos testes de tempo de resposta realizados para a comunicação desenvolvida nesse trabalho não considera-se a latência da rede. Para análises futuras mais aprofundadas pode-se incluir essa questão e também informações relacionadas ao hardware onde é executado o simulador, já que as suas configurações estão diretamente associadas a performance da execução das tarefas.

As pessoas que usam cadeiras de rodas elétricas enfrentam grandes barreiras quando se movimentam em lugares que não foram projetados para elas. Além disso, é preciso aumentar o conhecimento sobre as necessidades dessas pessoas com deficiência física, especialmente por pessoas responsáveis pelo desenvolvimento de ambientes públicos. Para esse fim, o simulador pode ser uma boa ferramenta para o aprofundamento nesse assunto e para ser usado como base para o design de ambientes.

O simulador e os cenários desenvolvidos neste trabalho podem ser utilizados como base para testes com outros tipos de interface de controle, visto que o Unity 3D apresenta a possibilidade de integrar vários dispositivos de entrada e saída.

Para dar continuidade aos estudos, pode-se também aplicar técnicas de inspeção de usabilidade do protótipo baseadas em entrevistas, simulações interativas e questionários dirigidos, para que assim, com base em informações, ideias e sugestões do próprio usuário possa ser desenvolvido um sistema que atenda às suas necessidades particulares e ainda lhe proporcione conforto e segurança. Com isso, será possível orientar e priorizar o trabalho de investigações das prováveis falhas de usabilidade que geram maiores impactos sobre a satisfação dos usuários.

## REFERÊNCIAS

- ALVARENGA, R. J. Reconhecimento de comandos de voz por redes neurais. Universidade de Taubaté, 2012.
- ANDROID DEVELOPER. **SpeechRecognizer Class**. [S.l.], 2022. Disponível em: <<https://developer.android.com/reference/android/speech/SpeechRecognizer>>.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT. NBR9050. **Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos**, v. 2, 2004.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT. NBR ISO 7176-3. **Cadeira de rodas Parte 3: Determinação da eficácia dos freios**, 2015.
- AZEVEDO, A.; WALCIMAR; EDUARDO, L. Interface homem-máquina com uso de comandos de voz embarcada em plataforma android para controle de dirigibilidade de uma cadeira de rodas motorizada. **XXI Congresso Brasileiro de Automática - CBA2016**, 2016.
- BYROM, B. et al. A review evaluating the validity of smartphone sensors and components to measure clinical outcomes in clinical research. **Value in Health**, Elsevier, v. 19, n. 3, p. A72, 2016.
- COOPER, R. A. et al. Comparison of virtual and real electric powered wheelchair driving using a position sensing joystick and an isometric joystick. **Medical engineering & physics**, Elsevier, v. 24, n. 10, p. 703–708, 2002.
- CORRÊA, D. M. et al. Ambiente virtual para simular a condução de uma cadeira de rodas. **Research, Society and Development**, v. 9, n. 1, p. e08911547–e08911547, 2020.
- DAY, T. W.; JOHN, N. W. Training powered wheelchair manoeuvres in mixed reality. In: IEEE. **2019 11th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)**. [S.l.], 2019. p. 1–7.
- DEVELOPEX. **Overview of Speech Recognition APIs for Android Platform**. [S.l.], 2022. Disponível em: <<https://developex.com/blog/overview-of-speech-recognition-apis-for-android-platform/>>.
- DEVMEDIA. **Como criar jogos: conheça as principais ferramentas**. [S.l.], 2023. Disponível em: <<https://www.devmedia.com.br/como-criar-jogos-conheca-as-principais-ferramentas/37848>>.
- DIAS, T. et al. **Interface Assistiva Para Interção com Ambiente Através de Movimento de Cabeça para Cadeirantes**. Tese (Doutorado) — Universidade Federal do Espírito Santo; Mestrado em Engenharia Elétrica , 2018.

GOMES, R. **Teste de Interfaces de Voz. 114 f.** Tese (Doutorado) — Dissertação—Faculdade de Engenharia da Universidade do Porto, Porto, 2007.

HAN, H.; BYUN, K.; KANG, H.-G. A deep learning-based stress detection algorithm with speech signal. In: **proceedings of the 2018 workshop on audio-visual scene understanding for immersive multimedia**. [S.l.: s.n.], 2018. p. 11–15.

HERRLICH, M. et al. Development of a virtual electric wheelchair—simulation and assessment of physical fidelity using the unreal engine 3. In: SPRINGER. **Entertainment Computing-ICEC 2010: 9th International Conference, ICEC 2010, Seoul, Korea, September 8-11, 2010. Proceedings 9**. [S.l.], 2010. p. 286–293.

MARQUES, L. S. et al. Finite element analysis of a commercial wheelchair. **Disability and Rehabilitation: Assistive Technology**, Taylor & Francis, v. 16, n. 8, p. 890–901, 2021.

MARQUES, L. S. et al. Virtual environment for smart wheelchair simulation. **IEEE Latin America Transactions**, IEEE, v. 19, n. 3, p. 456–465, 2021.

MARQUES, L. S. et al. Análise de tensões e deslocamentos em uma cadeira de rodas comercial via simulações numéricas. Universidade Federal de Lavras, 2019.

MATARNEH, R. et al. Speech recognition systems: A comparative review. IOSR, 2017.

MINISTÉRIO DA SAÚDE. **Diretrizes de Atenção à Pessoa com Lesão Medular**. [S.l.], 2013. Disponível em: <[http://bvsmms.saude.gov.br/bvsm/publicacoes/diretrizes\\_atencao\\_pessoa\\_lesao\\_medular.pdf](http://bvsmms.saude.gov.br/bvsm/publicacoes/diretrizes_atencao_pessoa_lesao_medular.pdf)>.

NASCIMENTO, D. F. d. **Modelagem de uma aplicação para dispositivos móveis de um planejador de rotas baseado em calçadas para usuários de cadeiras de rodas**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2018.

NETO, G. L. D. C.; LIMA, D. A. D.; NETO, A. D. M. The use of smartphones as a smart sensor for intelligent vehicles. In: IEEE. **2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)**. [S.l.], 2016. p. 37–42.

ORGANIZAÇÃO MUNDIAL DA SAÚDE (OMS). **Relatório Mundial sobre a Deficiência**. São Paulo, 2012.

PINTO, F. F. Interfaces de controle de cadeiras de rodas motorizadas para pessoas com tetraplegia. 2016.

PITHON, T. et al. Wheelchair simulators: A review. **Technology and Disability**, IOS Press, v. 21, n. 1-2, p. 1–10, 2009.

RABINER, L.; JUANG, B.-H. **Fundamentals of speech recognition**. [S.l.]: Prentice-Hall, Inc., 1993.

RODRIGO, S. E.; HERRERA, C. V. Wheelchairs: history, characteristics, and technical specifications. In: **Smart Wheelchairs and Brain-Computer Interfaces**. [S.l.]: Elsevier, 2008. p. 257–290.

SILVA, D. A. G. **Ausência de comunicação entre APP e AV**. 2023. Disponível em: <<https://youtube.com/shorts/SovG4lBuep0>>. Acesso em: 12 de fevereiro de 2023.

SILVA, D. A. G. **Código GitHub**. 2023. Disponível em: <[https://github.com/deboraags/tcc\\_d-bora.git](https://github.com/deboraags/tcc_d-bora.git)>. Acesso em: 16 de Março de 2023.

SILVA, D. A. G. **Experimento 1 - Demonstração do Funcionamento AV**. 2023. Disponível em: <<https://youtu.be/2EnThxtWwDE>>. Acesso em: 05 de fevereiro de 2023.

SILVA, D. A. G. **Experimento 2 - Demonstração do Funcionamento AV**. 2023. Disponível em: <<https://youtu.be/kpBo2yFoF7M>>. Acesso em: 05 de fevereiro de 2023.

SILVA, D. A. G. **Experimento 3 - Demonstração do Funcionamento AV**. 2023. Disponível em: <<https://youtu.be/y0C2qmgGCVo>>. Acesso em: 05 de fevereiro de 2023.

SILVA, Ê. dos S.; SEARA, R. Sistemas de reconhecimento automático de fala baseados em redes neurais profundas usando espectrogramas do sinal de fase. 2020.

SOARES, G.; LIMA, D. de; NETO, A. M. A mobile application for driver's drowsiness monitoring based on perclos estimation. **IEEE Latin America Transactions**, IEEE, v. 17, n. 02, p. 193–202, 2019.

UYANIK, C. et al. Brainy home: A virtual smart home and wheelchair control application powered by brain computer interface. In: **Biodevices**. [S.l.: s.n.], 2022. p. 134–141.