



UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Análise de Arquitetura baseada em nuvem utilizando protocolo MQTT para letramento em Braille

Dissertação de Mestrado

Josimar dos Santos



São Cristóvão – Sergipe

2023

**FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE**

Santos, Josimar dos
S237a Análise de Arquitetura baseada em nuvem utilizando protocolo
MQTT para letramento em braille / Josimar dos Santos ;
orientador Admilson de Ribamar Lima Ribeiro. - São Cristóvão,
2023.
82 f. : il.

Dissertação (mestrado em Ciência da Computação) –
Universidade Federal de Sergipe, 2023.

1. Internet das coisas. 2. Educação inclusiva. 3. Estudantes
com deficiência visual. 4. Computação em nuvem. 5. Rede de
computador – Protocolos. I. Ribeiro, Admilson de Ribamar Lima
orient. II. Título.

CDU 004.73

Josimar dos Santos

**Análise de Arquitetura baseada em nuvem utilizando protocolo
MQTT para letramento em Braille**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Sergipe como requisito para a obtenção do título de mestre em Ciência da Computação.

Trabalho aprovado.

São Cristóvão – Sergipe, 30 de Agosto de 2023:

Prof. Dr. Admilson de Ribamar Lima Ribeiro
Orientador

Prof. Dr. Gilton José Ferreira da Silva
Coorientador

Prof. Dr. Rubens de Souza Matos Junior
Interno a instituição

Prof. Dr. Jamilson Ramalho Dantas
Externo a instituição

São Cristóvão – Sergipe
2023

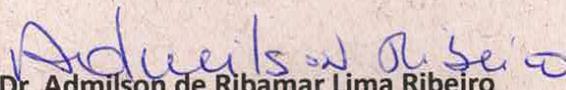


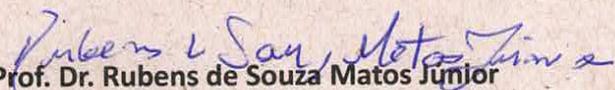
UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
COORDENAÇÃO DE PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

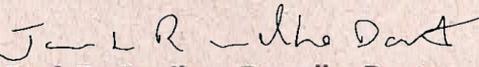
Ata da Sessão Solene de Defesa da Dissertação do
Curso de Mestrado em Ciência da Computação-UFS.
Candidato: **Josimar dos Santos**

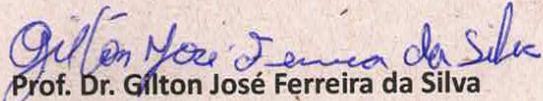
Em 30 dias do mês de agosto do ano de dois mil e vinte três, com início às 09h00min, realizou-se na Sala de Seminários do PROCC da Universidade Federal de Sergipe, na Cidade Universitária Prof. José Aloísio de Campos, a Sessão Pública de Defesa de Dissertação de Mestrado do candidato **Josimar dos Santos**, que desenvolveu o trabalho intitulado: "**Análise de Arquitetura baseada em nuvem utilizando protocolo MQTT para letramento em Braille**", sob a orientação do Prof. Dr. **Admilson de Ribamar Lima Ribeiro** e Dr. **Gilton José Ferreira da Silva**. A Sessão foi presidida pelo Prof. Dr. **Admilson de Ribamar Lima Ribeiro** (PROCC/UFS), que após a apresentação da dissertação passou a palavra aos outros membros da Banca Examinadora, Prof. Dr. **Jamilson Ramalho Dantas** (CIn-UFPE) e, em seguida, o Prof. Dr. **Rubens de Souza Matos Júnior** (Procc/UFS) e o Prof. Dr. **Gilton José Ferreira da Silva** (Procc/UFS). Após as discussões, a Banca Examinadora reuniu-se e considerou o mestrando (a) Aprovado "(aprovado/reprovado)". Atendidas as exigências da Instrução Normativa 05/2019/PROCC, do Regimento Interno do PROCC (Resolução 67/2014/CONEPE), e da Resolução nº 04/2021/CONEPE que regulamentam a Apresentação e Defesa de Dissertação, e nada mais havendo a tratar, a Banca Examinadora elaborou esta Ata que será assinada pelos seus membros e pelo mestrando.

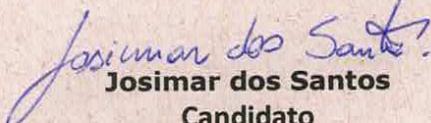
Cidade Universitária "Prof. José Aloísio de Campos", 30 de agosto de 2023.


Prof. Dr. **Admilson de Ribamar Lima Ribeiro**
(PROCC/UFS)
Presidente


Prof. Dr. **Rubens de Souza Matos Júnior**
(PROCC/UFS)
Examinador Interno


Prof. Dr. **Jamilson Ramalho Dantas**
(UFPE)
Examinador Externo


Prof. Dr. **Gilton José Ferreira da Silva**
(Procc/UFS)
Coorientador


Josimar dos Santos
Candidato

UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Josimar dos Santos

**Análise de Arquitetura baseada em nuvem utilizando protocolo
MQTT para letramento em Braille**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Sergipe como requisito para a obtenção do título de mestre em Ciência da Computação.

Orientador(a): Prof. Dr. Admilson de Ribamar Lima
Ribeiro

Coorientador(a): Prof. Dr. Gilton José Ferreira da Silva

São Cristóvão – Sergipe

2023

Dedico este trabalho a Deus, a minha Família e às pessoas com deficiência visual, buscando contribuir de alguma forma com ele para uma sociedade mais inclusiva e acessível. Com profundo respeito e admiração.

Agradecimentos

Primeiramente, expresso minha profunda gratidão a Deus, cuja bênção e orientação iluminaram minha trajetória, concedendo-me a força necessária para enfrentar os desafios e superar cada obstáculo com base na minha fé.

Quero agradecer imensamente minha amada família, um verdadeiro presente divino, que demonstrou uma incrível determinação e coragem ao enfrentar todas as dificuldades para me ajudar a crescer e me tornar a pessoa que sou hoje. Desde minha educação básica até a conquista do nível superior, e agora, me apoiando neste mestrado, sua dedicação foi fundamental. Em particular, agradeço minha querida avó, Maria Sabina (*in memoriam*), cujo apoio foi incondicional. Quero mencionar também minha tia, Rose Mère (*in memoriam*), que por tantas vezes substituiu minha mãe nas questões educacionais. Também, desejo honrar a memória de meu pai, Francisco Laurindo, que despertou em mim o desejo pelo conhecimento e a busca pela excelência acadêmica. Agradeço também à minha mãe, Joselita, minha irmã, Josiane Joselita, minha prima/irmã, Ranny Meirice, meu tio, Josemar Sebastião, Paulo Henrique, Pedro Henrique, Palmira Estéfany, Adjane Silva, Romário Souza, e os demais familiares, pela paciência, apoio e atenção constantes, pois me mostraram que a verdadeira união reside na harmonia de nossos laços. Gratidão à Iza Fontes, pelo apoio e orientação transformadores, proporcionando autoconhecimento e crescimento pessoal. Sua dedicação e escuta empática foram fundamentais para superar desafios. Gabriel e a Elane, meus sinceros agradecimentos pela força e incentivo, pois não há crescimento profissional se não houver, antes de tudo, uma estrutura educacional.

Expresso minha gratidão ao meu orientador, Prof. Dr. Admilson de Ribamar Lima Ribeiro, ao meu coorientador, Prof. Dr. Gilton José Ferreira da Silva, e aos dedicados professores que contribuíram para minha formação. Meus agradecimentos à Elaine Silva, secretária do Programa de Ciências da Computação (PROCC), que por inúmeras vezes, sempre solicita, colaborou com respostas às dúvidas que surgirem durante o caminho, e as mensagens de apoio e incentivo. Aos integrantes da banca internos, e ao Prof. Dr. Jamilson Ramalho Dantas, por darem suas significativas contribuições e aceitarem o convite.

Por último, agradeço a todos os colegas de mestrado, por compartilharem experiências e conhecimento. Aos que enfrentaram dificuldades e tiveram que interromper o percurso, encorajo-os a persistirem e nunca abandonarem seus sonhos. Àqueles que, apesar dos desafios, decidiram retomar os estudos, os incentivo a perseverarem. E para aqueles que concluíram esta jornada ao meu lado, antes e durante o mestrado, meus sinceros agradecimentos e votos de sucesso em todas as próximas etapas da vida.

*“Não basta saber, é preciso também aplicar.
Não basta querer, é preciso também agir.”
(Johann Goethe 1749-1832)*

Resumo

Na educação, a inclusão social, necessária para promover o desenvolvimento individual das Pessoas com Deficiência (PCD), muitas vezes requer a adaptação às necessidades individuais de cada aluno. Ao longo dos anos, avanços tecnológicos, técnicas e metodologias têm moldado essa realidade. A crescente utilização da Internet das Coisas (IoT) está conectando dispositivos e expandindo redes de sensores, permitindo a incorporação da acessibilidade no cotidiano das pessoas, incluindo a vida das PCD. Este trabalho pretende desenvolver uma abordagem inclusiva na educação, especialmente para PCD com deficiência visual. Propomos uma arquitetura baseada em nuvem, IoT e o protocolo *Message Queuing Telemetry Transport* (MQTT) para converter texto em Braille. Por meio dessa arquitetura em nuvem, buscamos simplificar a conversão de texto em Braille, promovendo acessibilidade e oferecendo uma experiência de aprendizado dinâmica e inclusiva para as PCD, com benefícios concretos para a comunidade educacional, incluindo a redução de custos. A estrutura utiliza o protocolo MQTT para estabelecer a comunicação entre dispositivos, permitindo a interação por meio de interfaces visuais e táteis, bem como comandos de voz. Além disso, o sistema opera em uma infraestrutura na nuvem, o que proporciona escalabilidade e facilita o acesso remoto. Após a conversão feita pela *Application Programming Interface* (API) na linguagem Python, um *JavaScript Object Notation* (JSON) é enviado para um simulador de display tátil, notificando o usuário por meio de sons que a leitura pode ser iniciada. Embora os experimentos realizados em ambiente local tenham obtido resultados iniciais satisfatórios, a validação do projeto exigirá a expansão para experimentos em situações reais, com um protótipo do display tátil.

Palavras-chave: Internet das coisas (IoT), educação, *e-Learning*, deficiente visual, pessoa com deficiência visual, computação em nuvem (nuvem), protocolo MQTT (MQTT), Braille, sistema Braille

Abstract

In education, social inclusion, necessary to promote the individual development of People with Disabilities (PWD), often requires adaptation to the individual needs of each student. Over the years, technological advancements, techniques, and methodologies have shaped this reality. The increasing use of the Internet of Things (IoT) is connecting devices and expanding sensor networks, enabling the integration of accessibility into people's daily lives, including the lives of PWD. This work aims to develop an inclusive approach to education, particularly for PWD with visual impairments. We propose a cloud-based architecture, IoT, and the Message Queuing Telemetry Transport (MQTT) protocol for converting text into Braille. Through this cloud-based architecture, our goal is to simplify text-to-Braille conversion, promoting accessibility and providing a dynamic and inclusive learning experience for PWD, with tangible benefits for the educational community, including cost reduction. The framework employs the MQTT protocol to establish communication between devices, enabling interaction through visual and tactile interfaces, as well as voice commands. Additionally, the system operates within a cloud infrastructure, offering scalability and facilitating remote access. After the conversion carried out by the Application Programming Interface (API) in the Python language, a JavaScript Object Notation (JSON) is sent to a tactile display simulator, notifying the user through sounds that reading can commence. While initial experiments conducted in a local environment yielded satisfactory results, project validation will require expansion to real-world experiments with a tactile display prototype.

Keywords: Internet of Things (IoT), education, e-Learning, visually impaired, person with visual impairment, cloud computing (cloud), MQTT protocol (MQTT), Braille, Braille system.

Lista de ilustrações

| | |
|--|----|
| Figura 1 – Pessoas que declararam algum tipo de deficiência no Brasil (Por Região) . . . | 22 |
| Figura 2 – Pesquisa Nacional por Amostra de Domicílios Contínua 2022 - PNAD Contínua | 24 |
| Figura 3 – Pessoas de 2 anos ou mais de idade com deficiência, segundo as Grandes Regiões e as Unidades da Federação (%) | 25 |
| Figura 4 – Reglete | 27 |
| Figura 5 – Escrita Braille: '4 DE JANEIRO DIA MUNDIAL DO Braille' | 28 |
| Figura 6 – Cega Braille | 29 |
| Figura 7 – Alfabeto Braille | 29 |
| Figura 8 – Numeral Braille | 30 |
| Figura 9 – Método de condução de um estudo de mapeamento sistemático | 37 |
| Figura 10 – Arquitetura Voice o Braille | 50 |
| Figura 11 – Arquitetura | 50 |
| Figura 12 – Simulador tinkercad: Modelagem de componentes | 51 |
| Figura 13 – Simulador tinkercad: Lista de componentes do modelo de teste | 52 |
| Figura 14 – Simulador tinkercad: Visão esquemática | 52 |
| Figura 15 – Simulador tinkercad: Codificação por blocos | 53 |
| Figura 16 – Simulador wokwi: Primeira conexão via MQTT | 53 |
| Figura 17 – Simulador wokwi: Recepção da mensagem após assinar no broker | 54 |
| Figura 18 – Simulador wokwi: Arduino ESP32 versão atual | 54 |
| Figura 19 – Mosquitto 2.0.15 - versão utilizada | 59 |
| Figura 20 – API: Tela com entrada de texto para conversão à esquerda e tela com texto "teste", convertido à direita. | 60 |
| Figura 21 – API: Envio do JSON por mensagem MQTT para ESP32 simulador | 60 |
| Figura 22 – Simulador wokwi: Arduino ESP32 em execução | 62 |
| Figura 23 – ABNT: Arranjo geométrico dos pontos em Braille | 65 |
| Figura 24 – ABNT: Formato do relevo do ponto em Braille | 66 |

Lista de quadros

| | |
|--|----|
| Quadro 1 – <i>String</i> utilizada para realizar as buscas nas bases | 39 |
|--|----|

Lista de tabelas

| | |
|---|----|
| Tabela 1 – Tipos de deficiências e seus respectivos níveis distribuídos por grupos e percentuais na população brasileira segundo o (IBGE, 2010) | 23 |
| Tabela 2 – Palavras-chave utilizadas na <i>string</i> de busca | 39 |
| Tabela 3 – Artigos por fonte | 40 |
| Tabela 4 – Artigos eliminados por Critérios de Exclusão [CE] | 40 |
| Tabela 5 – Artigos adicionados por Critérios de Inclusão [CI] | 41 |
| Tabela 6 – Artigos selecionados | 41 |
| Tabela 7 – Estudos Adicionados | 41 |
| Tabela 8 – Impressoras Braille e seus valores à vista | 46 |
| Tabela 9 – Impressoras Braille e seus preços em reais | 47 |
| Tabela 10 – Lista de Materiais e Custos | 47 |
| Tabela 11 – Material Permanente para o Projeto | 48 |
| Tabela 12 – Material de Consumo para a Bancada | 48 |
| Tabela 13 – Material de Consumo para o Circuito Elétrico e a Montagem Final | 49 |
| Tabela 14 – Descrição e Resultados | 51 |
| Tabela 15 – Ações Relacionadas Aos Botões Pressionados | 63 |
| Tabela 16 – Ações Relacionadas Aos Botões Pressionados | 63 |
| Tabela 17 – Frequência (Hz) Das Ações | 64 |

Lista de códigos

| | |
|---|----|
| Código 1 – Exemplo JSON | 31 |
| Código 2 – Código <i>Publish/Subscribe</i> | 61 |
| Código 3 – Código implementação <i>buzzer</i> | 63 |

Sumário

| | | |
|----------|---|-----------|
| 1 | Introdução | 16 |
| 1.1 | Apresentação Geral | 16 |
| 1.2 | Motivação | 18 |
| 1.3 | Justificativa | 18 |
| 1.4 | Objetivos | 19 |
| 1.4.1 | Objetivo Geral | 19 |
| 1.4.2 | Objetivo Específico | 19 |
| 1.4.3 | Contribuições esperadas | 20 |
| 1.5 | Metodologia | 21 |
| 1.6 | Estrutura do Documento | 21 |
| 2 | Fundamentação Teórica | 22 |
| 2.1 | Pessoas com Deficiência no Brasil | 22 |
| 2.1.1 | Deficiência Visual | 25 |
| 2.2 | Educação inclusiva | 27 |
| 2.2.1 | Educação inclusiva para deficientes visuais | 27 |
| 2.2.2 | Braille | 28 |
| 2.3 | Tecnologias da Informação e Comunicação (TIC) | 30 |
| 2.4 | Internet das Coisas (IoT) | 30 |
| 2.5 | JavaScript Object Notation (JSON) | 31 |
| 2.6 | Message Queuing Telemetry Transport (MQTT) | 32 |
| 2.7 | Computação em nuvem | 32 |
| 2.8 | Modelagem e impressão 3D e a cultura <i>Maker</i> | 34 |
| 2.9 | Arduino | 35 |
| 3 | Trabalhos Relacionados | 37 |
| 3.1 | Mapeamento sistemático | 37 |
| 3.1.1 | Questões de pesquisa | 38 |
| 3.1.2 | Estratégia de busca | 38 |
| 3.1.3 | Resultados | 40 |
| 3.1.4 | Resumos dos trabalhos | 41 |
| 3.1.5 | Discussões a respeito dos trabalhos | 42 |
| 3.1.5.1 | Tecnologias utilizadas | 42 |
| 3.1.5.2 | Braille e interação tátil | 43 |
| 3.1.5.3 | Assistência e comunicação | 43 |
| 3.1.5.4 | Arquiteturas e aplicabilidade | 44 |
| 3.1.5.5 | Benefícios relatados | 45 |
| 3.1.5.6 | Dificuldades e desafios | 45 |
| 4 | Análise e Arquitetura | 46 |
| 4.0.1 | Arquiteturas e aplicabilidade | 55 |
| 4.1 | Arquitetura | 57 |

| | | |
|----------|---|-----------|
| 5 | Resultados e Discussão | 58 |
| 5.1 | Detalhamento dos Experimentos | 58 |
| 5.2 | Trabalhos futuros | 67 |
| 6 | Considerações finais e Trabalhos Futuros | 68 |
| 6.1 | Publicação Realizada | 69 |
| 6.2 | Publicação Submetida | 69 |
| | Referências | 70 |
| | Anexos | 74 |
| | ANEXO A Artigo Publicado | 75 |

1

Introdução

Este capítulo estabelece uma contextualização concisa em relação às Pessoas com Deficiência, explorando seu contexto educacional e as abordagens pertinentes à proposta arquitetônica. São abordados tópicos como a motivação subjacente, a problemática abordada, a justificativa para o estudo e a estruturação do documento.

1.1 Apresentação Geral

As Pessoas com Deficiência (PCD) enfrentam desafios diários relacionados ao acesso e inclusão, apesar das garantias legais de equidade. A educação, como um pilar fundamental na formação do indivíduo, precisa ser revisada para garantir um processo de ensino alinhado com as necessidades singulares de cada aluno (COELHO; SOARES; ROEHRS, 2019).

De acordo com Coelho, Soares e Roehrs (2019), uma análise comparativa entre os Parâmetros Curriculares Nacionais (PCN) e a Base Nacional Comum Curricular (BNCC) em relação à inclusão de PCD na educação mostra que o PCN fornece mais orientações específicas para os professores, enquanto a BNCC, por ser mais normativa, assegurará uma educação igualitária para todos.

Além disso, as tecnologias estão transformando não apenas as tarefas do cotidiano das PCD, mas também no contexto educação. As Tecnologias da Informação e Comunicação (TIC) têm se consolidado na sociedade devido aos processos de globalização, trazendo mudanças significativas na forma como as pessoas consomem conteúdos digitais e interagem nas redes sociais. A conectividade possibilita que dispositivos se comuniquem e trabalhem em conjunto, impulsionando o desenvolvimento das redes de sensores da *Internet das Coisas* (IoT) (JUNIOR; VECE, 2016).

Diante desse cenário, este trabalho enfatiza a importância de uma educação inclusiva para pessoas com deficiência visual, que apresentam o maior índice de prevalência conforme o censo de 2010 IBGE (2010). O objetivo central é apresentar uma arquitetura em nuvem utilizando a *Internet das Coisas* (IoT) e o protocolo de comunicação *Message Queuing Telemetry Transport* (MQTT), que permita a conversão para o Sistema Braille e forneça uma saída compreensível para pessoas com deficiência visual. Essa iniciativa visa promover a inclusão e melhorias na qualidade de vida por meio da aplicação da tecnologia, prover acesso ao conhecimento de maneira mais

autônoma e independente para esse grupo específico de pessoas.

A presença e o impacto das Tecnologias da Informação e Comunicação (TIC) têm se consolidado cada vez mais na sociedade, especialmente impulsionadas pelos avanços da globalização. Essas tecnologias trouxeram consigo mudanças significativas e características marcantes na forma como as pessoas consomem conteúdos digitais e interagem nas redes sociais. A conectividade avançada possibilitou a comunicação e colaboração entre dispositivos, impulsionando o desenvolvimento das redes de sensores da *Internet das Coisas* (IoT).

O dispositivo tátil pode se comunicar com a infraestrutura na nuvem por meio da IoT para receber informações convertidas em Braille e notificar o usuário de que a leitura pode ser iniciada. Essa comunicação bidirecional entre o dispositivo tátil e a nuvem é um exemplo de como a IoT pode ser usada para tornar a experiência de aprendizado mais inclusiva e dinâmica para pessoas com deficiência visual.

O MQTT desempenha o papel de facilitar a comunicação entre os componentes do sistema, permitindo que dispositivos IoT, a infraestrutura na nuvem e outros componentes se comuniquem de maneira eficiente e escalável. Isso é fundamental para o funcionamento adequado do projeto, garantindo que as mensagens sejam entregues no momento certo aos dispositivos e usuários relevantes.

Além disso, a construção dessa arquitetura pode ser de grande valia para a comunidade educacional e a sociedade em geral, ao oferecer soluções tecnológicas acessíveis e inclusivas para o aprendizado em sala de aula do Braille. A aplicação de tecnologias em nuvem e a utilização do protocolo MQTT podem proporcionar uma experiência de leitura e aprendizado mais dinâmica e interativa para os usuários.

Dessa forma, a arquitetura tem o potencial de impactar positivamente a vida de muitas pessoas com deficiência visual, permitindo que elas acessem o conhecimento de maneira mais autônoma e independente. Ao integrar o mundo físico com o mundo digital, essa iniciativa representa mais um passo em direção a uma sociedade inclusiva e igualitária, onde a tecnologia é utilizada como uma ferramenta para promover o acesso à educação e o pleno desenvolvimento de todas as pessoas, independentemente de suas habilidades visuais. O presente estudo está estruturado da seguinte maneira: inicialmente, iremos elucidar os conceitos teóricos fundamentais, construídos a partir de estudos preexistentes. Neste contexto, iremos detalhar os processos inerentes à metodologia de pesquisa adotada neste estudo. A arquitetura será minuciosamente apresentada, com uma ênfase na descrição dos seus elementos primordiais e das suas funções respectivas. Além disso, ofereceremos uma análise detalhada dos resultados obtidos mediante a aplicação da mencionada arquitetura. Exploraremos, de maneira abrangente, possíveis direções para investigações futuras, visando aprofundar ainda mais o entendimento nesse domínio específico. No decorrer desta exposição, serão compartilhadas as reflexões pertinentes a esse estudo, destacando os principais discernimentos gerados pelo projeto. Tais considerações serão seguidas pelas conclusões alcançadas a partir das análises realizadas. Finalmente, será fornecida uma lista completa das fontes que forneceram embasamento e fundamentação para o desenvolvimento deste estudo, reforçando a sua validade e robustez.

1.2 Motivação

A educação inclusiva tem se tornado um tema fundamental na agenda educacional, buscando proporcionar um ambiente de aprendizagem adaptado às necessidades individuais de cada aluno, incluindo aqueles com deficiência visual (COELHO; SOARES; ROEHRS, 2019). No entanto, apesar das legislações e das iniciativas para garantir a igualdade de acesso à educação, muitas pessoas com deficiência visual ainda enfrentam obstáculos significativos que prejudicam sua participação plena na sociedade, como o alto custo de equipamentos necessários (MARTILLANO et al., 2018).

Nesse sentido, a Internet das Coisas (IoT) tem se mostrado uma solução tecnológica promissora não apenas em empresas, mas também em áreas como o governo e a educação. A IoT oferece um enorme potencial para o desenvolvimento de soluções inovadoras e inclusivas na área educacional, especialmente para pessoas com deficiência visual (MARTILLANO et al., 2018).

Com base nessa demanda, esta pesquisa propõe a modelagem de uma arquitetura em nuvem, utilizando IoT e o protocolo MQTT, que ofereça uma saída compreensível para pessoas com deficiência visual. Essa arquitetura pretende promover a inclusão educacional e social, fornecendo às pessoas com deficiência visual acesso autônomo à aprendizagem do Braille. Além disso, o modelo busca explorar o potencial da tecnologia para criar um ambiente educacional mais flexível, adaptado e inclusivo.

Ao combinar conhecimentos em tecnologia com uma compreensão das necessidades específicas das pessoas com deficiência visual, esta pesquisa tem o potencial de impactar positivamente a vida de muitas pessoas, promovendo sua autonomia, independência e igualdade de oportunidades. Por meio de uma abordagem científica e inovadora, espera-se contribuir para a criação de um ambiente educacional mais inclusivo, onde a tecnologia seja utilizada como uma ferramenta para promover o acesso ao conhecimento para todos, independentemente de suas habilidades visuais. Essa pesquisa busca avançar no conhecimento científico e tecnológico, oferecendo benefícios sociais e educacionais relevantes para a comunidade acadêmica e a sociedade na totalidade (MARTILLANO et al., 2018).

1.3 Justificativa

A presente pesquisa se fundamenta na conjunção entre a crescente disponibilidade de tecnologias de conectividade e a necessidade premente de promover a inclusão de pessoas com deficiência visual no contexto da educação. Atualmente, vivenciamos uma era em que dispositivos interconectados desempenham papéis cruciais nas esferas de entretenimento, aprendizagem e negócios empresariais. No entanto, as barreiras impostas pela deficiência visual representam um desafio significativo para a efetiva participação desses indivíduos nesse cenário digital em constante evolução.

A deficiência visual requer abordagens de ensino que se utilizem dos sentidos do tato, da audição e até mesmo do olfato, a fim de suplantar a ausência ou limitação do sentido da visão. Em ambientes educacionais tradicionais, as exposições visuais, como imagens, vídeos e textos, tornam-se obstáculos substanciais para o aprendizado e a inclusão dessas pessoas. Isso, por sua vez, contribui para altos índices de analfabetismo entre esse grupo, privando-o de oportunidades de desenvolvimento e participação plena na sociedade.

Nesse contexto, a pesquisa se propõe a explorar soluções tecnológicas inovadoras e inclusivas que direcionem os sentidos disponíveis dos alunos com deficiência visual. A ideia central é proporcionar uma experiência de ensino-aprendizado mais acessível, dinâmica e interativa, permitindo a esses alunos compartilhar conhecimentos e habilidades por meio das ferramentas tecnológicas e digitais amplamente disponíveis. A contribuição deste estudo não se limita a um contexto meramente acadêmico, mas estende-se ao âmbito social, uma vez que almeja mitigar a desigualdade educacional e facilitar a inclusão de pessoas com deficiência visual em ambientes cada vez mais tecnológicos.

Além disso, o trabalho se insere em um contexto de evolução constante das pesquisas relacionadas à conversão do Sistema Braille para a era digital. Com base em trabalhos como o de [Santiago et al. \(2020\)](#), que propõe a criação de um display tátil do Sistema Braille conectado via Universal Serial Bus (USB), e [Ramos-García et al. \(2022\)](#), que desenvolve estudos correlatos, mas em um contexto mais recente, o projeto visa aprimorar e expandir as abordagens de conectividade para a conversão de texto em Braille.

Outras iniciativas, como o projeto de [Torres e Ramirez \(2019\)](#), demonstram que soluções como o display Braille são viáveis, porém, ainda demandam discussões e melhorias significativas em termos de conectividade e conversão.

Portanto, a pesquisa proposta visa aprofundar o conhecimento nesse campo, explorando novas formas de conectividade e aprimorando a acessibilidade da educação para pessoas com deficiência visual, contribuindo assim para a promoção de uma sociedade mais inclusiva e equitativa. A justificativa deste estudo reside no imperativo de enfrentar os desafios existentes e criar soluções eficazes para a inclusão digital de pessoas com deficiência visual, capacitando-as para participar plenamente do mundo cada vez mais conectado em que vivemos.

1.4 Objetivos

Esta seção apresentará o Objetivo Geral deste trabalho, bem como os Objetivos Específicos.

1.4.1 Objetivo Geral

- Projetar uma arquitetura em nuvem utilizando o protocolo MQTT para letramento em Braille.

1.4.2 Objetivo Específico

Com o objetivo geral definido, para reunir e alinhar conhecimentos em torno do problema, os seguintes objetivos específicos foram estabelecidos:

- **Compreender as necessidades das Pessoas com Deficiência visual e seus desafios no contexto da aprendizagem**, visando desenvolver soluções acessíveis e inclusivas.
- **Realizar revisão da literatura sobre o Sistema Braille** para embasar o projeto, abordando origens, evolução e padrões de codificação.

- **Investigar a IoT e computação em nuvem**, abrangendo protocolos, dispositivos, sensores, plataformas e segurança para construir uma arquitetura eficiente.
- **Testar ferramentas práticas** relacionadas ao tema.
- **Projetar a arquitetura** para simular a arquitetura objetivo deste estudo.
- **Documentar o desenvolvimento da arquitetura e resultados** para compartilhar conhecimentos e contribuir com tecnologias assistivas.

1.4.3 Contribuições esperadas

Este estudo pretende oferecer uma série de contribuições significativas, entretanto alguns aspectos serão discutidos e analisados em trabalhos futuros. As contribuições esperadas incluem:

- **Gerenciamento, Autenticação e Segurança Adaptados:** O projeto buscará desenvolver soluções de gerenciamento de informações e autenticação que levem em consideração as necessidades específicas dos alunos com deficiência visual. Isso inclui o desenvolvimento de métodos de autenticação seguros e de fácil acesso, bem como a proteção das informações pessoais do aluno.
- **Implementação Física Aprimorada:** Serão exploradas melhorias na implementação física do dispositivo tátil Braille, visando torná-lo mais acessível, durável e fácil de usar. A ergonomia e a portabilidade do dispositivo serão consideradas.
- **Protótipos de Display Físicos:** A pesquisa pretende criar protótipos de displays táteis físicos que atendam às necessidades dos alunos com deficiência visual. Isso envolve a exploração de diferentes designs e a escolha dos materiais mais adequados.
- **Melhoria da API e Servidor:** A API usada para a conversão de texto em Braille será aprimorada para maior eficiência e precisão. Além disso, a pesquisa visa disponibilizar a API em um servidor acessível, facilitando a integração com outros sistemas educacionais.
- **Atividades Dinâmicas de Interação:** O estudo buscará criar atividades interativas que envolvam o reconhecimento tátil e a fala. Isso promoverá uma experiência de aprendizado dinâmica, envolvente e inclusiva para os alunos.
- **Testes de Vibração:** Serão realizados testes de vibração no dispositivo em vez do uso de um buzzer para notificação. Isso visa proporcionar uma experiência sensorial mais agradável e precisa para os alunos com deficiência visual.
- **Integração da Assistente de Voz no Display Braille:** A pesquisa explorará a possibilidade de implementar a assistente de voz diretamente no dispositivo tátil Braille, levando em consideração a acessibilidade e o baixo custo do projeto. Isso permitirá interações mais naturais e eficazes.
- **Testes de Usabilidade:** Serão conduzidos testes de usabilidade para avaliar a eficácia e a aceitação do sistema pelos alunos com deficiência visual. Os resultados dos testes orientarão melhorias adicionais no projeto.

1.5 Metodologia

Este trabalho de pesquisa foi classificado de acordo com [Silva e Menezes \(2005\)](#), tendo suas características descritas da seguinte forma:

- **Quanto à natureza:** trata-se de uma pesquisa aplicada, com o propósito de produzir conhecimento voltado para a aplicação prática de uma arquitetura, visando resolver problemas e atender interesses comerciais e científicos.
- **Quanto à abordagem do problema:** A pesquisa adota uma abordagem qualitativa devido ao uso do processo de mapeamento sistemático, que seleciona estudos relevantes ao tema abordado para fornecer embasamento teórico e identificar lacunas exploradas.
- **Quanto aos objetivos:** A pesquisa tem objetivos exploratórios e descritivos, buscando uma maior familiarização com o problema por meio de pesquisas bibliográficas e exploração dos recursos mais utilizados nas soluções propostas em trabalhos relacionados. A intenção é analisar e testar esses recursos, modelando uma arquitetura que permita a conversão de texto para voz, voz para texto, em nuvem, e por meio de uma API própria. A arquitetura deve possibilitar a transmissão dessa conversão, em algum momento, por meio do protocolo MQTT para um *display* tátil, fornecendo uma saída personalizada de caracteres convertidos em Braille.
- **Quanto aos procedimentos técnicos:** trata-se de uma pesquisa bibliográfica, devido à utilização de materiais já publicados e disponíveis através da *Internet*. Esse levantamento teórico foi fundamental para o desenho da arquitetura que é o principal objetivo deste trabalho.

1.6 Estrutura do Documento

Este documento está organizado em capítulos e seções, abrangendo os seguintes tópicos:

- Capítulo 1 - Introdução: contextualiza o tema proposto, apresentando motivação, justificativa e objetivos deste trabalho.
- Capítulo 2 - Fundamentação teórica: explora o contexto teórico relacionado ao tema por meio de revisão de literatura.
- Capítulo 3 - Trabalhos relacionados: apresenta os trabalhos correlatos e os resultados do mapeamento sistemático e revisão de literatura adotada, bem como a síntese do processo de sumarização dos trabalhos estudados.
- Capítulo 4 - Proposta de Dissertação: detalha a arquitetura.
- Capítulo 5 - Considerações Finais: encerra o documento com as conclusões e considerações finais do trabalho.

A estrutura do documento foi organizada para proporcionar uma apresentação coerente e clara do trabalho realizado, abordando os aspectos relevantes do tema proposto.

2

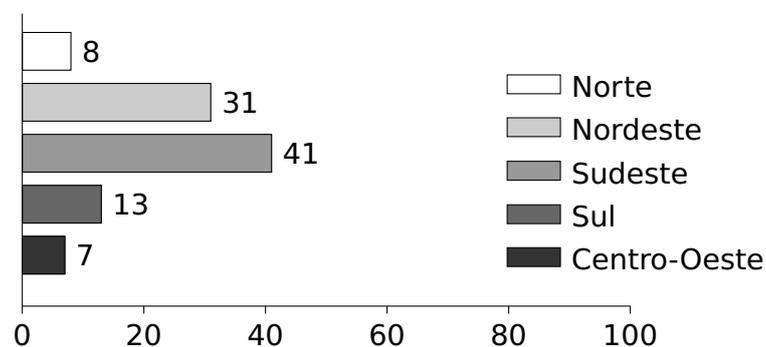
Fundamentação Teórica

Neste capítulo, abordaremos assuntos relacionados a pessoas com deficiência no Brasil, destacando dificuldades visuais e discutiremos a Educação Inclusiva. Apresentaremos o Sistema Braille, além de discutir tópicos como: TIC, IoT, JSON, protocolo MQTT, computação em nuvem, e a cultura maker que enfatiza a utilização da impressão 3D e o uso do Arduino na criação de dispositivos.

2.1 Pessoas com Deficiência no Brasil

A deficiência, conforme a Lei Brasileira de Inclusão da Pessoa com Deficiência (Lei nº 13.146/2015), [Brasil \(2015\)](#), é compreendida como uma limitação corporal, sensorial, cognitiva ou psicológica que, em interação com variadas barreiras, pode dificultar a plena participação e integração social da pessoa. A legislação busca assegurar direitos, equidade de oportunidades e a eliminação de preconceitos, promovendo a acessibilidade em todas as esferas da vida.

Figura 1 – Pessoas que declararam algum tipo de deficiência no Brasil (Por Região)



Fonte: ([IBGE, 2010](#))

Com base nos dados do Censo Demográfico de 2010, conduzido pelo Instituto Brasileiro de Geografia e Estatística (IBGE) [IBGE \(2010\)](#), é possível observar os índices de pessoas que relataram ter algum tipo de deficiência. A distribuição desses índices por região no Brasil revela

uma visão panorâmica, destacando o Nordeste e o Sudeste com as taxas mais elevadas, de acordo com [IBGE \(2010\)](#):

Nesta análise da [Figura 1](#), demonstra de forma abrangente as taxas de pessoas que se declaram possuir algum tipo de deficiência em diferentes regiões do Brasil. Chama a atenção a significativa representação das regiões Nordeste e Sudeste, com as mais altas proporções, conforme documentado pelo Censo de 2010 ([IBGE, 2010](#)).

No [Tabela 1](#) apresenta os tipos de deficiências e seus respectivos níveis distribuídos por grupos e percentuais na população brasileira, segundo o Censo Demográfico de 2010 ([IBGE, 2010](#)).

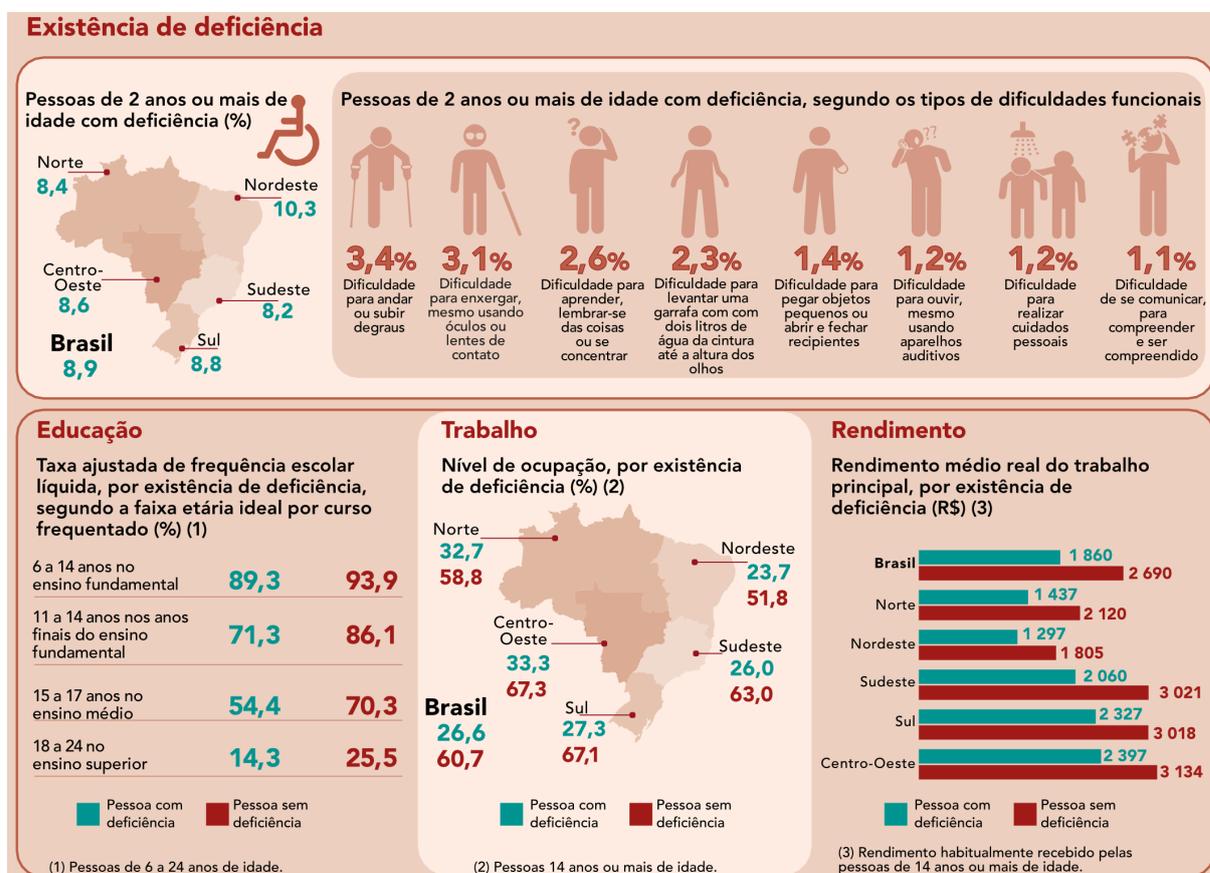
Ainda no [Tabela 1](#) apresenta os diferentes tipos de deficiências e seus respectivos níveis distribuídos por grupos e percentuais na população brasileira, conforme dados coletados no Censo Demográfico de 2010 ([IBGE, 2010](#)). O total da população brasileira na época do censo era de 190.755.799 pessoas, e 23.92% dessa população, correspondendo a 45.623.910 pessoas, declararam possuir pelo menos uma das deficiências investigadas.

Tabela 1 – Tipos de deficiências e seus respectivos níveis distribuídos por grupos e percentuais na população brasileira segundo o ([IBGE, 2010](#))

| Tipo de deficiência Brasil | | |
|--|--------------------|--------------|
| Total população brasileira Censo 2010 - 190.755.799 | | |
| Tipo/Nível | População | % |
| Pelo menos uma das deficiências investigadas | 45.623.910 | 23.92 |
| Deficiência visual — não consegue de modo algum | 528.624 | 0.28 |
| Deficiência visual — grande dificuldade | 6.056.684 | 3.18 |
| Deficiência visual — alguma dificuldade | 29.206.180 | 15.31 |
| Deficiência visual total | 35.791.488 | 18.76 |
| Deficiência auditiva — não consegue de modo algum | 347.481 | 0.18 |
| Deficiência auditiva — grande dificuldade | 1.799.885 | 0.94 |
| Deficiência auditiva — alguma dificuldade | 7.574.797 | 3.97 |
| Deficiência auditiva total | 9.722.163 | 5.10 |
| Deficiência motora — não consegue de modo algum | 740.456 | 0.39 |
| Deficiência motora — grande dificuldade | 3.701.790 | 1.94 |
| Deficiência motora — alguma dificuldade | 8.831.723 | 4.63 |
| Deficiência motora total | 13.273.969 | 6.96 |
| Mental/intelectual | 2.617.025 | 1.37 |
| Nenhuma dessas deficiências | 145.084.578 | 76.06 |
| Sem declaração | 47.311 | 0.02 |

Fonte: ([IBGE, 2010](#))

Figura 2 – Pesquisa Nacional por Amostra de Domicílios Contínua 2022 - PNAD Contínua



Fonte: Pesquisa Nacional por Amostra de Domicílios Contínua 2022, Instituto Brasileiro de Geografia e Estatística (IBGE) (2023)

Dentre as deficiências investigadas, destaca-se a deficiência visual, que possui três níveis: não consegue de modo algum, grande dificuldade e alguma dificuldade. O total de pessoas com alguma deficiência visual representa 18.76% da população brasileira, correspondendo a 35.791.488 pessoas.

Vale ressaltar, que devido o problema da pandemia do COVID-19, novas pesquisas foram adiadas, mas conforme Instituto Brasileiro de Geografia e Estatística (IBGE) (2023), em que a primeira parte inclui o informativo com os principais resultados da pesquisa, disponível em formato impresso e digital no portal do IBGE. A segunda parte é constituída pelas Notas técnicas, abordando aspectos metodológicos, acessíveis somente em formato digital no portal do IBGE. O Brasil possui 18,6 milhões de pessoas (8,9%) com deficiência, dentre aquelas com 2 anos ou mais de idade, em relação à população total de 203.062.512 em 2022. Na Figura 2 podemos observar os dados da pesquisa. Nesta pesquisa, a dificuldade para enxergar, mesmo usando óculos ou lentes de contato, entra como segunda posição com 3,1%, ficando atrás apenas da dificuldade para andar ou subir degraus, com 3,4%, Figura 3.

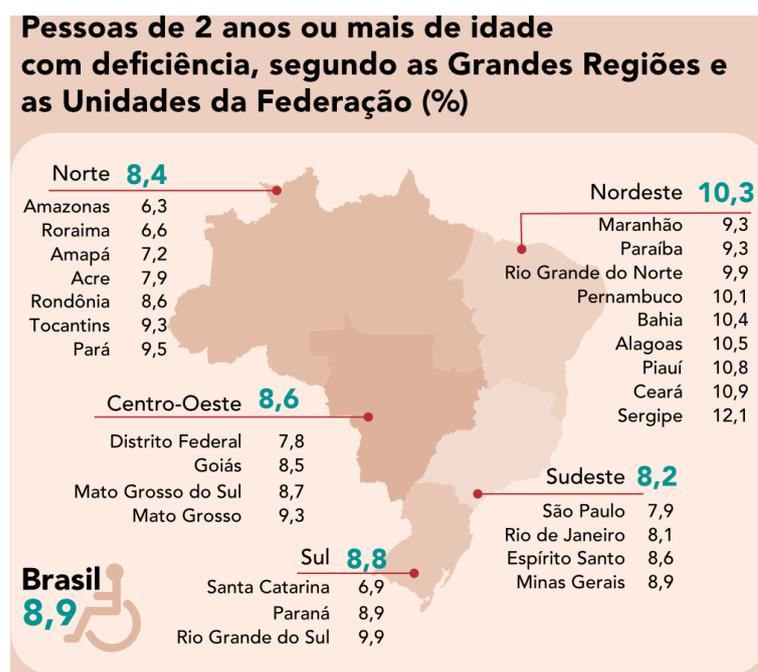
Conforme a PNAD Contínua de 2022, Instituto Brasileiro de Geografia e Estatística (IBGE) (2023), a estimativa é de 18,6 milhões de indivíduos com deficiência no Brasil, que

representa 8,9% da população com 2 anos ou mais. Nesse grupo, o perfil é mais prevalente no gênero feminino (10,0%) em comparação com o masculino (7,7%). Quanto à etnia, as pessoas autodeclaradas como pretas (9,5%) têm maior incidência, em comparação a pardas (8,9%) e brancas (8,7%), [Figura 3](#).

Essa prevalência apresentou variações regionais, sendo o Nordeste com 5,8 milhões de pessoas com deficiência, liderando com 10,3% da população - acima da média nacional. Todas as Unidades da Federação na região tiveram valores acima da média, com destaque para Sergipe (12,1%), como mostra [Figura 3](#). As outras Grandes Regiões apresentaram percentuais próximos: Sudeste (8,2%), Norte (8,4%), Centro-Oeste (8,6%) e Sul (8,8%). No contexto das Unidades da Federação, o Amazonas exibiu o menor percentual, com 6,3% da população possuindo deficiência ([Instituto Brasileiro de Geografia e Estatística \(IBGE\), 2023](#)).

Considerando os dados das últimas pesquisas, [Figura 1](#) e [Figura 3](#), a região Nordeste continua ocupando uma posição com os maiores índices de pessoas com deficiência no Brasil.

Figura 3 – Pessoas de 2 anos ou mais de idade com deficiência, segundo as Grandes Regiões e as Unidades da Federação (%)



Fonte: Pesquisa Nacional por Amostra de Domicílios Contínua 2022, [Instituto Brasileiro de Geografia e Estatística \(IBGE\) \(2023\)](#)

2.1.1 Deficiência Visual

A deficiência visual pode ser congênita ou adquirida e pode afetar a capacidade de identificar cores, tamanhos, distâncias, formas, posições e interagir com o ambiente, afetando a mobilidade do indivíduo. Pessoas com deficiência visual tendem a utilizar outros sentidos para captar informações do ambiente, como o tato e a audição ([ESTEVEVES et al., 2019](#)).

A legislação brasileira, especificamente o decreto n.º 5.296 de 2 de dezembro de 2004 em

Brasil (2004), classifica a deficiência visual em dois níveis: cegueira, quando a acuidade visual é igual ou menor que 0,05 no melhor olho com a melhor correção óptica; e baixa visão, quando a acuidade visual está entre 0,3 e 0,05 no melhor olho com a melhor correção óptica. Além disso, também considera a ocorrência simultânea dessas condições ou quando a somatória da medida do campo visual em ambos os olhos for igual ou menor que 60°. Outras leis que abordam direitos e condições gerais de acessibilidade PCDs: Brasil (2015) e a Brasil (2000).

No contexto da deficiência visual, podem existir circunstâncias congênitas ou adquiridas que afetam a capacidade das pessoas em identificar cores, tamanhos, distâncias, formas e posições, além de comprometer a interação com o ambiente e afetar a mobilidade do indivíduo. Diante dessa situação, as pessoas com deficiência visual tendem a utilizar outros sentidos sensoriais, como o tato e a audição, para captar estímulos e informações do ambiente (ESTEVES et al., 2019).

A legislação visa garantir os direitos das pessoas com deficiência, incluindo a acessibilidade em espaços públicos e privados, bem como o acesso à informação e ao conhecimento. Entretanto, a ausência de recursos acessíveis pode impossibilitar a mobilidade, autonomia e independência dos deficientes, resultando em exclusão e isolamento social. Portanto, é fundamental que a arquitetura e o urbanismo se adéquem às necessidades das pessoas com deficiência visual, proporcionando uma maior inclusão e acessibilidade em todas as esferas da sociedade (BRASIL, 2004).

A legislação brasileira, por meio do Decreto n.º 5.296 de 2 de dezembro de 2004 Brasil (2004), classifica a deficiência visual em diferentes níveis:

- Cegueira: acuidade visual igual ou menor que 0,05 no melhor olho, com a melhor correção óptica;
- Baixa visão: acuidade visual entre 0,3 e 0,05 no melhor olho, com a melhor correção óptica;
- Casos nos quais a somatória da medida do campo visual em ambos os olhos é igual ou menor que 60°;
- Ocorrência simultânea de quaisquer das condições anteriores.

Os níveis de acuidade visual e campo de visão podem ser definidos por profissionais capacitados, como oftalmologistas, por meio de testes e exames específicos que possibilitam identificar as características da visão em cada caso.

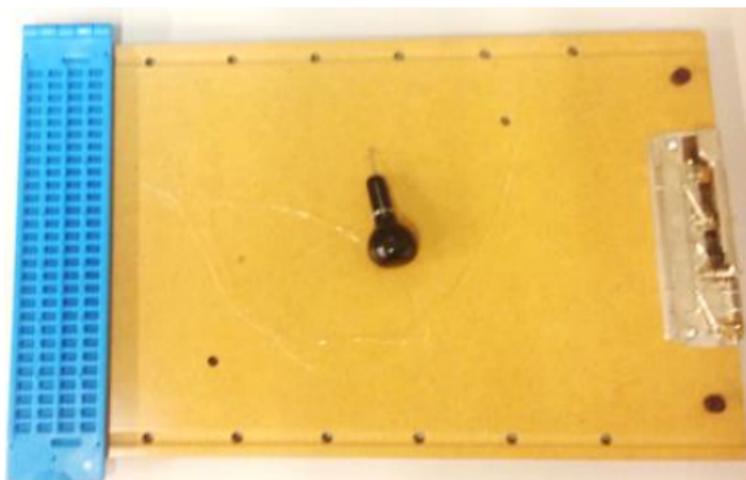
A legislação, deve garantir os direitos das pessoas com deficiência e estabelece que a arquitetura e o urbanismo devem se adequar às necessidades dessas pessoas, visando promover a acessibilidade e a inclusão social. Entretanto, a ausência de recursos acessíveis nos espaços públicos e privados pode impossibilitar a mobilidade, o acesso à informação e ao conhecimento, bem como a autonomia e independência dos deficientes, resultando em isolamento e exclusão desse grupo considerável da população (BRASIL, 2004). Portanto, é fundamental que as políticas públicas e as iniciativas privadas se comprometam a implementar medidas que garantam a acessibilidade e a inclusão das pessoas com deficiência visual e de outras deficiências, contribuindo para uma sociedade mais justa e igualitária.

2.2 Educação inclusiva

A escola desempenha um papel fundamental não apenas no desenvolvimento cognitivo dos alunos, mas também na socialização e no enriquecimento humano. Quando se trata de educação inclusiva, é essencial criar um ambiente que estimule o processo de desenvolvimento de todos os alunos, incluindo aqueles com necessidades especiais (COELHO; SOARES; ROEHRS, 2019).

Diversas metodologias, técnicas e recursos tecnológicos, tanto digitais como analógicos, são essenciais para promover a inclusão, capacitar os alunos e facilitar o acesso à informação e ao conhecimento de maneira didática. Entre essas ferramentas, destacam-se o Braille, Tangram, Soroban, Reglete na Figura 4, Livros em Braille, MEC Daisy, livros falados, Máquina de Escrever em Braille, Impressora Braille e Sintetizadores de voz. Nesta subseção, abordaremos especificamente o Braille para uma compreensão mais aprofundada dessa arquitetura.

Figura 4 – Reglete

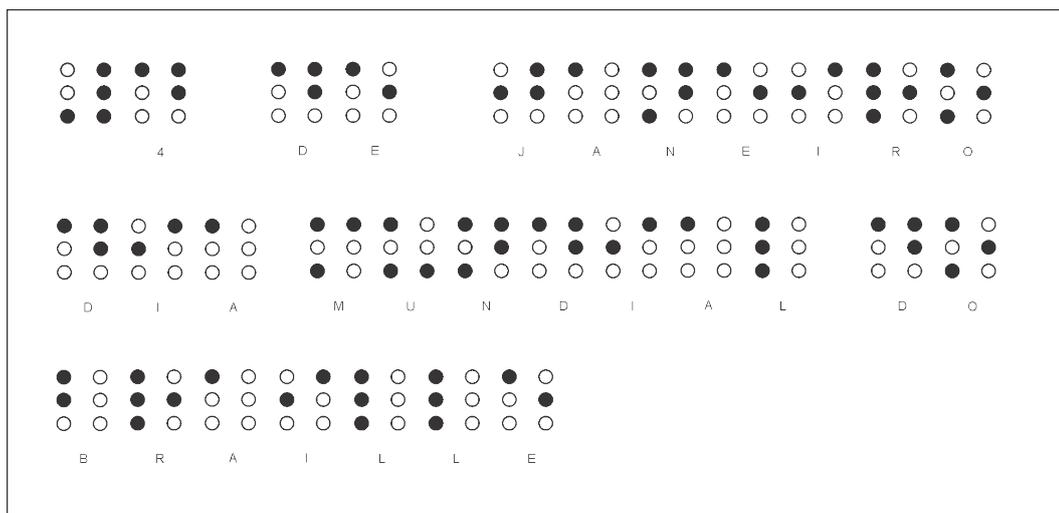


Fonte: Santiago et al. (2020)

2.2.1 Educação inclusiva para deficientes visuais

Conforme mencionado por Gandhi, Thakker e Jha (2017), a ausência total ou parcial da visão pode dificultar o processo de aprendizagem, como a leitura e escrita, levando a altas taxas de analfabetismo entre os deficientes visuais. No entanto, os autores destacam a existência de uma linguagem especial, criada em 1824 por Louis Braille, destinada a esse grupo de pessoas, conhecida como Braille (confira a Figura 5).

Figura 5 – Escrita Braille: '4 DE JANEIRO DIA MUNDIAL DO Braille'



Fonte: Autores (2021)

O Braille é um sistema tátil de escrita e leitura baseado em símbolos em relevo, que permite que pessoas com deficiência visual possam ler e escrever com autonomia. Cada símbolo Braille é composto por uma combinação de seis pontos dispostos em duas colunas verticais de três pontos cada. Esses pontos são dispostos em várias combinações, representando letras, números, símbolos matemáticos e até mesmo notas musicais.

A inclusão do Braille nas práticas educacionais é essencial para garantir o acesso pleno à informação e ao conhecimento para os deficientes visuais. Além disso, é importante que educadores e instituições estejam preparados para oferecer suporte e incentivo aos alunos com deficiência visual, promovendo a igualdade de oportunidades e a participação ativa na vida acadêmica. Para alcançar uma educação inclusiva de qualidade, é necessário utilizar recursos como o Braille, bem como outras técnicas e tecnologias, adaptando o ensino para atender às necessidades individuais de cada aluno. Somente assim, poderemos construir uma sociedade mais justa e acessível para todos.

2.2.2 Braille

Para compreendermos como ocorrerá o processo de conversão e exibição no *display* Braille, é essencial entendermos o próprio sistema Braille.

Segundo [Gandhi, Thakker e Jha \(2017\)](#), o sistema Braille é baseado em células ou celas formadas por uma matriz de pontos em alto-relevo, disposta em um formato de três linhas e duas colunas, conforme representado na [Figura 6](#). Cada célula Braille possui seis pontos numerados de 1 a 6, e a combinação entre pontos erguidos ou não forma representações de letras, números e símbolos para mapeamento tátil. Dessa forma, cada célula Braille representa um caractere específico.

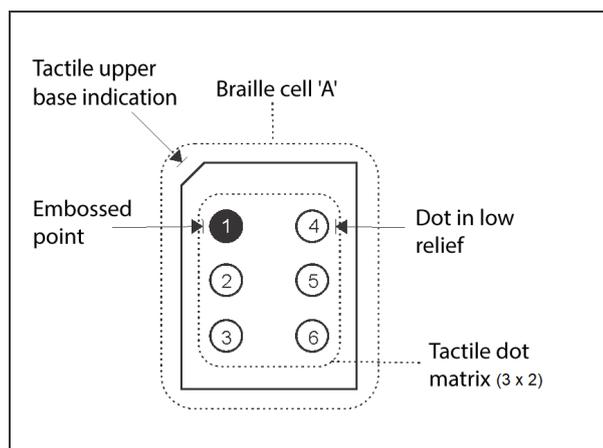
As células Braille são dispostas lado a lado para formar palavras e textos completos, proporcionando às pessoas com deficiência visual uma leitura tátil e acessível. O Alfabeto Braille, apresentado na [Figura 7](#), mostra as diferentes combinações de pontos que representam as letras

do nosso alfabeto. O mesmo princípio se aplica aos números, onde a única variação relevante neste contexto é a presença de um símbolo precursor antes do próprio símbolo numérico. Este padrão Braille que antecede o número é composto pelos pontos [3456] elevados e pelos pontos [12] rebaixados, formando a imagem inversa da letra "L", como ilustrado na [Figura 8](#).

No contexto deste projeto, o objetivo é permitir que o usuário envie comandos de voz ao sistema, que serão convertidos em texto e, em seguida, traduzidos para o formato Braille. Essa conversão de texto para Braille será exibida no *display* Braille, que apresentará os caracteres em alto ou baixo-relevo nas células Braille, possibilitando a leitura tátil por parte das pessoas com deficiência visual.

Dessa forma, ao compreendermos o sistema Braille e suas representações táteis, podemos visualizar como o dispositivo poderá viabilizar a leitura tátil para o usuário, proporcionando uma experiência de aprendizado inclusiva e acessível a pessoas com deficiência visual, bem como àqueles interessados em aprender o Braille. O projeto busca, assim, empregar a tecnologia para promover a inclusão e a igualdade no processo educacional.

Figura 6 – Cella Braille



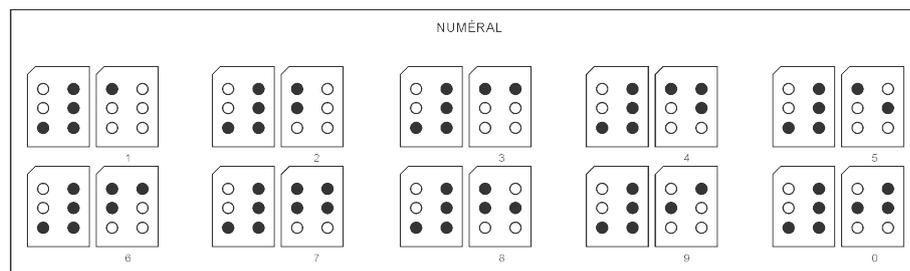
Fonte: Autores (2021)

Figura 7 – Alfabeto Braille

| ALPHABET | | | | | | | | | | | | |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| A | B | C | D | E | F | G | H | I | J | K | L | M |
| ● ○ | ● ○ | ● ● | ● ● | ● ○ | ● ● | ● ● | ● ○ | ○ ● | ○ ● | ● ● | ● ○ | ● ● |
| ○ ○ | ○ ○ | ○ ○ | ○ ● | ○ ● | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ |
| ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ |
| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| ● ● | ● ○ | ● ● | ● ● | ● ○ | ○ ● | ○ ● | ● ○ | ● ○ | ○ ● | ○ ● | ○ ● | ○ ○ |
| ○ ● | ○ ● | ○ ● | ○ ● | ○ ● | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ | ○ ○ |
| ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ | ● ○ |

Fonte: Autores (2021)

Figura 8 – Numeral Braille



Fonte: Autores (2021)

2.3 Tecnologias da Informação e Comunicação (TIC)

O autor [Possato e Monteiro \(2020\)](#), destaca a relevância das Tecnologias da Informação e Comunicação (TIC) no âmbito educacional, ressaltando que o desenvolvimento científico e tecnológico tem sido uma constante ao longo da história humana. [Possato e Monteiro \(2020\)](#) (2020 apud Kenski 2007) afirma que “as tecnologias são tão antigas quanto à espécie humana”, o que enfatiza a importância de refletir sobre a relação entre educação e tecnologia. Nesse contexto, é essencial considerar como a sociedade humana tem sido moldada pela presença da tecnologia.

Essa evolução tecnológica é crucial não apenas para os seres humanos, mas também para o desenvolvimento do pensamento e para a criação de artefatos tecnológicos que desempenham um papel fundamental no cotidiano das pessoas. Essas tecnologias conseguem auxiliar em atividades diárias, facilitar a comunicação e também melhorar o processo de ensino-aprendizagem.

Ao longo da história, a educação tem sido fortemente influenciada pelas transformações socioeconômicas da sociedade. O crescente uso das tecnologias proporcionou novas possibilidades de interação, comunicação e aprendizagem. É importante destacar que a construção do conhecimento humano está intrinsecamente relacionada à interação com outras pessoas, uma vez que a identidade de cada indivíduo é historicamente construída e situada no contexto social ([ATAÍDE; MESQUITA, 2014](#)).

[Ataíde e Mesquita \(2014\)](#), complementa a discussão, enfatizando a rápida expansão do mercado de tecnologia, o que acaba por influenciar a inserção dessas tecnologias nos ambientes educacionais. Esse cenário impulsiona a busca por soluções educacionais mais inovadoras e alinhadas às necessidades contemporâneas, permitindo o aproveitamento das vantagens que as TIC podem oferecer para o aprimoramento da educação e o desenvolvimento das habilidades dos estudantes.

2.4 Internet das Coisas (IoT)

Segundo [Lins e Morais \(2021\)](#), a Internet das Coisas (IoT) é definida como um sistema em que dispositivos inteligentes estão interconectados, permitindo a cooperação entre eles. Esses dispositivos têm acesso a diversos dados dos usuários, que precisam ser gerenciados eficientemente para garantir a segurança das informações. Nesse contexto, a tecnologia *Blockchain* é apresentada como uma opção promissora, ao permitir o registro de dados de forma descentralizada, criptografada e imutável. O uso do *Blockchain* pode tornar a IoT mais segura,

uma vez que os dados não ficam centralizados em um único servidor, tornando mais difícil para hackers comprometerem a integridade das informações.

A computação ubíqua é uma abordagem que tornará a interatividade entre usuários e dispositivos algo natural, imperceptível e onipresente. É uma evolução da computação tradicional, onde os dispositivos estão interligados e trabalham proativamente para fornecer serviços de maneira transparente aos usuários. Essa tecnologia tem aplicação em diversas áreas, como Medicina, Educação, Infraestrutura, negócios e sociedade em geral. Assim como a *Internet* revolucionou ao conectar computadores globalmente, a IoT surge como uma nova revolução ao integrar uma infinidade de dispositivos inteligentes, impactando profundamente como os seres humanos interagem e interpretam o mundo ao seu redor (PATRICIO et al., 2018).

2.5 JavaScript Object Notation (JSON)

JSON (*JavaScript Object Notation*) é um formato leve de intercâmbio de dados amplamente utilizado na web e em aplicações modernas. Foi introduzido como parte da especificação ECMAScript em sua 5ª edição e tem ganhado popularidade devido à sua simplicidade e facilidade de uso (RAMOS-GARCÍA et al., 2022).

Concebido como um formato de texto, o JSON é facilmente legível tanto para humanos como para máquinas. Ele utiliza uma estrutura de pares de chave-valor, onde os dados são organizados em objetos e arrays. Cada objeto é delimitado por chaves “{ }” e contém um conjunto de pares chave-valor separados por dois pontos “:”. Já os arrays são delimitados por colchetes “[]” e podem conter uma lista ordenada de valores, incluindo outros objetos e arrays (ECMA International, 2017). Um exemplo simples de JSON é o seguinte:

Código 1 – Exemplo JSON

```
1 {  
2   "nome": "Joao",  
3   "idade": 30,  
4   "profissao": "Engenheiro",  
5   "interesses": ["leitura", "viagens", "culinaria"]  
6 }
```

Fonte: Autores (2023)

Nesse exemplo, [Código 1](#), temos um objeto que representa uma pessoa chamada Joao. O objeto contém as chaves “nome”, “idade”, “profissao” e “interesses”, cada uma associada a um valor específico.

Segundo o [ECMA International \(2017\)](#), JSON é amplamente utilizado em aplicações web para enviar e receber dados entre o cliente e o servidor. Além disso, muitas APIs (*Interfaces de Programação de Aplicativos*) utilizam JSON como formato de troca de informações, tornando a integração de diferentes sistemas mais fácil e eficiente.

Com a sua simplicidade e a facilidade de análise e geração por parte de linguagens de programação, o JSON se tornou uma escolha popular para o transporte de dados estruturados em muitas aplicações e serviços modernos. Sua adoção crescente e a sua ampla documentação

fazem dele uma ferramenta fundamental para o desenvolvimento de sistemas interconectados e flexíveis na era digital (ECMA International, 2017).

2.6 Message Queuing Telemetry Transport (MQTT)

O protocolo MQTT, desenvolvido pela IBM em 1999, é uma solução de comunicação baseada em um sistema de publicação/assinatura que utiliza o protocolo TCP para garantir a segurança e transferência eficiente de dados (YASSEIN et al., 2017). Especificamente projetado para operar em condições de longo atraso de rede e baixa largura de banda, o MQTT se destaca por sua capacidade de enviar dados com precisão e confiabilidade.

Neste projeto, a API (Application Programming Interface) desempenha um papel fundamental ao receber os comandos do usuário, os quais são fornecidos em formato JSON (JavaScript Object Notation), um formato compacto amplamente utilizado para troca de informações. Quando o comando “converter” é recebido pela API, ela emite uma mensagem contendo os dados relevantes e publica essa mensagem no broker MQTT.

O broker MQTT, por sua vez, é um intermediário que gerencia a comunicação entre a API e o display Braille.

Segundo Santiago et al. (2020), O display Braille é um dispositivo que pode ser descrito como um periférico tátil de saída de computador, possibilitando a formação autônoma de arranjos de pontos empregados para facilitar a interpretação no contexto do sistema Braille. Ele recebe as mensagens publicadas pela API e as encaminha para o display Braille, que está inscrito em um tópico específico para receber essas mensagens. O display Braille, então, interpreta as informações recebidas e cria uma representação tátil do conteúdo, permitindo que o usuário com deficiência visual compreenda o conteúdo apresentado.

Essa integração eficiente entre a API e o protocolo MQTT é de extrema importância para o sucesso do projeto, ao garantir a transmissão segura e rápida dos dados, possibilitando que o usuário com deficiência visual tenha acesso ao conteúdo em Braille de forma ágil e confiável. Por meio dessa solução tecnológica, o projeto visa proporcionar uma experiência de aprendizado inclusiva, promovendo a participação ativa de pessoas com deficiência visual no processo educacional. Além disso, o uso do MQTT como protocolo de comunicação oferece uma solução prática e confiável para a conversão de texto em Braille, contribuindo significativamente para a inclusão e acessibilidade de informações às pessoas com deficiência visual.

Através dessa iniciativa, busca-se o avanço na democratização do conhecimento, valorizando a igualdade de oportunidades e promovendo a inclusão digital, permitindo que todos os indivíduos, independentemente de suas habilidades visuais, tenham acesso pleno à informação e educação.

2.7 Computação em nuvem

A computação em nuvem é uma abordagem revolucionária que tem em vista oferecer uma série de benefícios significativos para usuários e empresas. Entre esses benefícios, destaca-se a disponibilidade contínua dos serviços, a mobilidade para acessar os recursos de qualquer lugar, a flexibilidade para ajustar a capacidade de processamento e armazenamento conforme a necessidade, a segurança robusta para proteger os dados e informações sensíveis, o compartilhamento

de recursos entre diferentes usuários e a redução de custos em infraestrutura física (CALANCEA et al., 2019).

Esses avanços na computação em nuvem são possíveis graças a técnicas de virtualização, que permitem a criação de ambientes virtuais independentes em servidores físicos, e ao modelo de negócio “pague-pelo-que-use”, no qual os usuários pagam somente pelos recursos que realmente utilizam (SPOHN; BONETTI, 2019). Essa abordagem traz uma economia significativa para empresas, que podem evitar gastos excessivos com infraestrutura subutilizada.

Além disso, a infraestrutura na nuvem facilita a escalabilidade dos recursos, o que significa que as aplicações podem facilmente lidar com aumento de demanda sem a necessidade de investimentos em hardware adicional. Isso é especialmente benéfico em cenários de picos de tráfego, como em vendas online durante eventos especiais (SPOHN; BONETTI, 2019).

No contexto do projeto em questão, a computação em nuvem possibilita a implementação de diversas aplicações relevantes. Por exemplo, a hospedagem de aplicação em nuvem permite que os serviços estejam disponíveis na *Internet*, acessíveis de qualquer dispositivo conectado (SPOHN; BONETTI, 2019). O armazenamento em nuvem oferece uma solução escalável para guardar grandes volumes de dados, garantindo a disponibilidade e redundância dos mesmos.

No âmbito do reconhecimento de voz (fala-para-texto) e da conversão de texto-para-fala, a computação em nuvem disponibiliza poderosos algoritmos e recursos para realizar essas tarefas de maneira eficiente e precisa. Essas capacidades são essenciais para permitir a comunicação e interação com o usuário por meio de interfaces de voz, o que é especialmente relevante no contexto de acessibilidade para pessoas com deficiência visual.

Ademais, a utilização de brokers em nuvem para gerenciar as publicações e subscrições do protocolo MQTT é uma solução prática e escalável para controlar a troca de mensagens entre dispositivos em um ambiente de *Internet* das Coisas (IoT). Isso é fundamental para garantir a comunicação eficiente e segura entre os dispositivos conectados, permitindo o funcionamento integrado do sistema no projeto.

Apesar dos inúmeros benefícios, é importante ressaltar que alguns serviços em nuvem oferecem um período gratuito para testes, mas em alguns casos, o uso gratuito é limitado. Conforme a demanda do projeto e suas necessidades específicas, pode ser necessário formalizar a contratação de planos pagos oferecidos por empresas líderes em serviços de nuvem, como Google (Google Cloud) ¹, Amazon (AWS) ², Microsoft (Microsoft Azure) ³ e IBM (IBM Cloud) ⁴.

Essas empresas têm investido significativamente em infraestrutura e recursos, garantindo alta disponibilidade, confiabilidade e segurança em seus serviços de computação em nuvem.

Em suma, a computação em nuvem apresenta-se como uma base sólida para o desenvolvimento do projeto, proporcionando soluções escaláveis, eficientes e economicamente viáveis para a implementação da arquitetura e para a oferta de serviços de conversão de texto para voz, voz para texto e outras funcionalidades em nuvem, atendendo às necessidades dos usuários e promovendo a inclusão e acessibilidade para pessoas com deficiência visual.

¹ <<https://cloud.google.com/>>

² <<https://aws.amazon.com/pt/>>

³ <<https://azure.microsoft.com/pt-br/>>

⁴ <<https://cloud.ibm.com/>>

2.8 Modelagem e impressão 3D e a cultura *Maker*

A cultura *maker* desempenha um papel fundamental na *Internet* das Coisas (IoT), ao fornecer um ambiente propício para a criação e experimentação de dispositivos e soluções conectadas. Na cultura *maker*, a abordagem colaborativa e criativa incentiva as pessoas a explorarem tecnologias emergentes, como a impressora 3D, para prototipação e desenvolver dispositivos personalizados e adaptados às suas necessidades específicas (SANTIAGO et al., 2020).

Com a utilização da impressora 3D e outras ferramentas disponíveis na cultura *maker*, as pessoas podem construir de forma independente e econômica os componentes necessários para seus projetos de IoT. Torres e Ramirez (2019), em seu projeto de desenvolvimento de uma cela Braille, ou basicamente, um *display* Braille, demonstra resultados de como pode se economizar na estrutura desta ideia com a utilização de impressão 3D por meio de filamento. Isso permite que desenvolvam dispositivos sob medida, otimizados para suas aplicações particulares, e que consigam resolver problemas específicos de forma eficiente e inovadora (SANTIAGO et al., 2020).

Além disso, a cultura *maker* valoriza o compartilhamento de conhecimento e experiência entre os seus membros. Mediante plataformas online e comunidades locais de *makers*, as pessoas podem trocar ideias, soluções e aprimoramentos, acelerando o desenvolvimento e a adoção de tecnologias IoT. Essa colaboração contínua contribui para um ecossistema de IoT mais rico e diversificado, com uma ampla gama de soluções e abordagens (SANTIAGO et al., 2020).

Santiago et al. (2020) afirma que a cultura *maker* também promove a aprendizagem ativa e a resolução de problemas práticos, incentivando as pessoas a serem criadoras e protagonistas em vez de apenas consumidoras passivas de tecnologia. Essa abordagem *hands-on* é essencial para o desenvolvimento de habilidades técnicas e criativas, capacitando os indivíduos a enfrentarem desafios e explorarem novas possibilidades na IoT.

Portanto, a cultura *maker* é uma peça-chave no cenário da IoT, impulsionando a inovação, a acessibilidade e a democratização das tecnologias conectadas. Ela capacita as pessoas a aproveitarem ao máximo o potencial da impressora 3D e outras ferramentas de prototipagem rápida para criar soluções sob medida, contribuindo para um futuro tecnológico mais colaborativo e inclusivo. A integração da modelagem e da impressão 3D neste projeto é fundamental para a criação de um produto final, mesmo que se trate de um protótipo, com uma aparência mais refinada e funcional. Através da técnica de modelagem e da tecnologia de impressão 3D, será possível desenvolver uma estrutura personalizada, como um *case*, que proporcionará um acabamento adequado, ocultando fios, microcontroladores e peças utilizadas.

Além disso, a impressão 3D viabiliza a construção das partes mecânicas necessárias para a implementação dos pontos em relevo utilizados no sistema Braille. Essas partes podem ser projetadas com precisão e adaptadas às especificidades do projeto, garantindo um funcionamento otimizado e eficiente.

Com a utilização da modelagem 3D, é possível criar modelos virtuais detalhados do dispositivo antes de imprimir fisicamente, permitindo uma visualização prévia do produto e facilitando ajustes e correções, caso necessário. Essa abordagem iterativa torna o processo de desenvolvimento mais ágil e eficaz, reduzindo o tempo e os recursos necessários para a criação do projeto (SANTIAGO et al., 2020).

A impressão 3D também oferece vantagens em termos de flexibilidade e personalização, ao permitir a produção de peças únicas e adaptadas às necessidades específicas do projeto. Isso é especialmente relevante para este estudo, que tem em vista criar uma arquitetura baseada em nuvem utilizando protocolo MQTT para letramento em Braille. A capacidade de personalização oferecida pela impressão 3D permitirá adaptar o dispositivo às particularidades do usuário e às exigências da leitura em Braille (SANTIAGO et al., 2020).

Em resumo, a integração da modelagem e impressão 3D no projeto proporcionará uma estrutura funcional e esteticamente agradável, otimizando a experiência do usuário e contribuindo para a eficácia e sucesso do sistema de letramento em Braille. A tecnologia de impressão 3D enriquece a concepção do dispositivo, garantindo um resultado mais profissional e adequado às necessidades específicas do público-alvo.

2.9 Arduino

No âmbito da cultura *maker* e da prototipagem de produtos, o Arduino desponta como uma tecnologia amplamente adotada para construir dispositivos eletrônicos e computacionais. Criado visando facilitar a prototipagem rápida de projetos de computação física, o Arduino se tornou uma ferramenta versátil e acessível para entusiastas, estudantes e profissionais (SANTIAGO et al., 2020).

Segundo Santiago et al. (2020), a tecnologia Arduino é composta por três elementos principais, que contribuem para sua popularidade e flexibilidade:

Software: O Arduino oferece uma plataforma de desenvolvimento que permite a criação e programação de códigos para controlar e interagir com os dispositivos conectados. Por meio de uma linguagem de programação baseada em *Wiring* e uma interface gráfica amigável, os usuários podem desenvolver facilmente programas para seus projetos.

Legalidade: A comunidade do Arduino promove uma abordagem colaborativa e de código aberto, o que significa que o uso da tecnologia não está sujeito a restrições de direitos autorais. Isso estimula a disseminação do conhecimento e incentiva a troca de ideias e projetos entre os membros da comunidade *maker*.

Hardware: As placas Arduino são o coração da tecnologia e vêm em diversas variações e tamanhos, adaptadas para diferentes tipos de projetos e necessidades. Cada placa é equipada com componentes eletrônicos, como processador, memória, pinos de entrada e saída, que permitem a interação com sensores, atuadores e outros dispositivos.

Essa diversidade de placas e a disponibilidade de uma ampla gama de componentes eletrônicos permitem que os usuários personalizem seus projetos de acordo com suas necessidades específicas. Torres e Ramirez (2019), por exemplo, utilizou uma placa do Arduino Uno para seu projeto. Além disso, a comunidade Arduino é extremamente ativa, compartilhando projetos, tutoriais e bibliotecas de código aberto, agilizando o processo de desenvolvimento e aprendizado (SANTIAGO et al., 2020).

Dessa forma, o Arduino se tornou uma peça-chave na cultura *maker* e na prototipagem de produtos, capacitando indivíduos a transformar suas ideias em realidade de forma rápida e acessível. Sua combinação de software, legalidade e hardware acessível oferece uma base sólida para a criação de projetos inovadores e estimula a criatividade e a colaboração na comunidade

maker. A tecnologia Arduino se destaca como uma ferramenta essencial para impulsionar a inovação e a experimentação no mundo da eletrônica e da computação física (SANTIAGO et al., 2020).

3

Trabalhos Relacionados

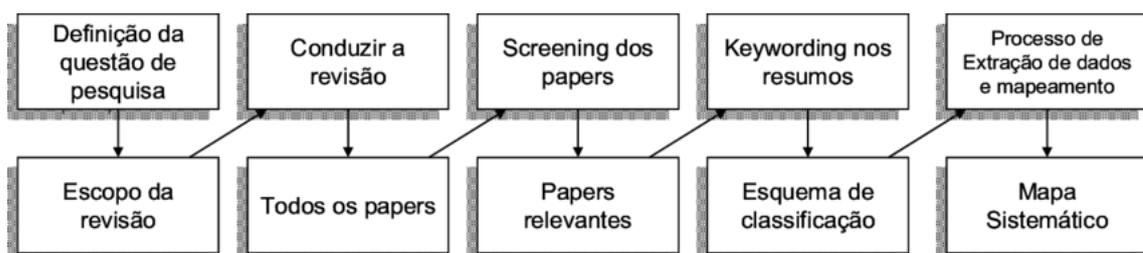
Neste capítulo, são reunidos os processos utilizados para a obtenção dos resultados, por meio de um mapeamento sistemático realizado entre setembro de 2021 a dezembro de 2021.

3.1 Mapeamento sistemático

De acordo com [Kitchenham, Budgen e Brereton \(2011\)](#), a metodologia com base no mapeamento sistemático envolve o planejamento de uma pesquisa visando mapear toda a literatura de estudos empíricos e não empíricos em uma determinada área temática. Esse mapeamento permite a contabilização, seleção, qualificação e, por fim, a extração dos dados relevantes para responder não só à questão principal, mas também a questões derivadas dela. O processo pode ser dividido em cinco etapas: **1** - Definição de questões de pesquisa; **2** - Condução da Pesquisa de Estudos Primários; **3** - Triagem de artigos com base nos critérios de inclusão/exclusão; **4** - Classificação dos artigos; e **5** - Extração e agregação de dados.

O mesmo planejamento se aplica a esta pesquisa, tendo como condução principal a pergunta base: “A implementação de uma IoT aplicada na educação para pessoas com deficiência visual pode se tornar uma aliada para os atores em questão?”. Ao se deparar com esta pergunta, outras seis perguntas foram formuladas e apresentadas na [subseção 3.1.1](#) deste trabalho. Para visualizar o planejamento de forma esquemática, pode-se dividir este trabalho em fases representadas na [Figura 9](#).

Figura 9 – Método de condução de um estudo de mapeamento sistemático



3.1.1 Questões de pesquisa

Esta pesquisa tem o objetivo de responder às seguintes questões:

- Q1** Quais tecnologias específicas de *Internet das Coisas* (IoT) estão sendo atualmente aplicadas na educação inclusiva de pessoas com deficiência visual? Quais os principais dispositivos e sistemas utilizados?
- Q2** Como as tecnologias de IoT são estruturadas para atender às necessidades da educação inclusiva de pessoas com deficiência visual? Quais as principais características e funcionalidades consideradas nesse contexto?
- Q3** De que forma a implementação da IoT pode proporcionar conforto, autonomia e segurança física específicos para pessoas com deficiências visuais na área educacional? Quais os benefícios percebidos por essas pessoas?
- Q4** Considerando as particularidades da deficiência visual, como as tecnologias de IoT são adaptadas para abordar questões de acessibilidade, interação e usabilidade para os usuários com perda parcial ou total da visão?
- Q5** Quais são os principais desafios e dificuldades enfrentados na aplicação de IoT na educação inclusiva para pessoas com deficiência visual? Quais as limitações e obstáculos enfrentados pelos pesquisadores e desenvolvedores?
- Q6** Quais são os principais fóruns acadêmicos e meios de divulgação onde os trabalhos relacionados à IoT para educação inclusiva são publicados? Quais as fontes mais relevantes de informação nessa área?

3.1.2 Estratégia de busca

Foram utilizadas as seguintes bases de pesquisa:

- Scopus¹;
- IEEE Xplore Digital Library²;
- Web of Science³;
- Science Direct⁴;

Na [Tabela 2](#), são apresentadas as Palavras-Chave utilizadas para formar a *string* de busca. A *string* de busca, [Quadro 1](#), é composta por palavras-chave e operadores lógicos, tais como os "AND" e "OR" que constituíram a expressão utilizada na pesquisa.

¹ <<http://www.scopus.com>>

² <<http://ieeexplore.ieee.org>>

³ <<https://www.webofknowledge.com/>>

⁴ <<http://www.sciencedirect.com>>

Tabela 2 – Palavras-chave utilizadas na *string* de busca

| Palavra-chave | Sinônimo em inglês |
|--------------------------------------|--------------------------------------|
| <i>Internet das coisas</i> | <i>Internet of Things, IoT</i> |
| <i>educação, e-learning</i> | <i>education, e-learning</i> |
| <i>deficiente visual</i> | <i>visually impaired</i> |
| <i>pessoa com deficiência visual</i> | <i>people with visually impaired</i> |
| <i>computação na nuvem</i> | <i>cloud computing, cloud</i> |
| <i>protocolo MQTT</i> | <i>MQTT protocol, MQTT</i> |
| <i>Sistema Braille</i> | <i>Braille System</i> |

Fonte: Autores (2021)

Quadro 1 – *String* utilizada para realizar as buscas nas bases

((*Internet of things OR iot*) AND (*education OR e-learning*) AND (*cloud computing OR cloud*) AND (*visually impaired OR people with visually impaired*) AND (*mqtt protocol OR mqtt*) AND (*barille system*))

Fonte: Autores (2021)

O acesso às bases de dados foi realizado por meio do portal de periódicos da [CAPES \(2018\)](#), utilizando as credenciais de acesso fornecidas pela Universidade Federal de Sergipe (UFS). Dessa forma, obtivemos a possibilidade de fazer o download completo das publicações disponíveis.

A identificação dos estudos primários foi conduzida através da aplicação de uma *string* de busca nas bases selecionadas. Essa *string* de busca consiste em palavras-chave e operadores lógicos, como os conectivos “AND” e “OR”, que foram combinados para compor a estratégia de busca empregada neste Mapeamento Sistemático.

A seguir os Critérios de Inclusão [CI] enumerados:

- CI1** — Relevância com o tema proposto;
- CI2** — Artigos, conferências ou periódicos;
- CI3** — Número de citações maior ou igual a 1;

E estes os Critérios de Exclusão [CE] enumerados:

- CE1** — Estudos duplicados;
- CE2** — Publicações de 2011 aos mais atuais.
- CE3** — Não disponíveis, incompletos ou corrompidos.

3.1.3 Resultados

A pesquisa foi realizada aplicando as *string* de busca nas bases relacionadas na [subseção 3.1.2](#), sem filtro inicial através das *string* de busca apresentada no [Quadro 1](#). Neste primeiro momento, após descarte de estudos duplicados, obtivemos o resultado como mostra a [Tabela 3](#).

Tabela 3 – Artigos por fonte

| Base | Estudos retornados |
|-----------------------------|--------------------|
| Scopus | 10 |
| IEEE Xplore Digital Library | 9 |
| Web of Science | 10 |
| Science Direct | 59 |
| Total | 88 |

Fonte: Autores (2021)

Para descartar os artigos duplicados, utilizamos a ferramenta [parsif.al](#)⁵ e classificamos os artigos conforme os critérios de inclusão e exclusão mencionados na [subseção 3.1.2](#). A seleção dos artigos foi baseada, prioritariamente, nos títulos e palavras-chave. Em seguida, analisamos os resumos e, por fim, o texto completo para identificar os artigos relevantes.

No processo de classificação, foram selecionados artigos publicados em 2011 aos mais atuais.

Tabela 4 – Artigos eliminados por Critérios de Exclusão [CE]

| Base | Critérios de Exclusão [CE] |
|-----------------------------|----------------------------|
| Scopus | [CE1:2][CE2:0][CE3:0] |
| IEEE Xplore Digital Library | [CE1:3][CE2:0][CE3:0] |
| Web of Science | [CE1:4][CE2:0][CE3:0] |
| Science Direct | [CE1:0][CE2:0][CE3:0] |
| Total | 9 |

Fonte: Autores (2021)

O processo de extração prossegue com um total de 88 artigos e aplicai-se os critérios de inclusão [CI] e exclusão [CE]. Na [Tabela 4](#), temos as identificações por critério de exclusão e as respectivas quantidades de estudos excluídos. De 88 estudos, 9 não foram selecionados na primeira etapa da seleção. Restaram 79 artigos submetidos aos critérios com base aos Critérios de Inclusão [CI], conforme [Tabela 5](#), apenas 4 estudos foram selecionados. Nesta etapa, todos os artigos foram lidos completamente para checagem de relevância com conteúdo abordado.

⁵ <<https://parsif.al>>

Tabela 5 – Artigos adicionados por Critérios de Inclusão [CI]

| Base | Critérios de Inclusão [CI] |
|-----------------------------|----------------------------|
| Scopus | [CI1:2][CI2:0][CI3:0] |
| IEEE Xplore Digital Library | [CI1:0][CI2:0][CI3:0] |
| Web of Science | [CI1:0][CI2:0][CI3:0] |
| Science Direct | [CI1:2][CI2:0][CI3:0] |
| Total | 4 |

Fonte: Autores (2021)

Na [Tabela 6](#), detalha quais foram os 4 trabalhos selecionados, bases de pesquisa retornada, seus respectivos autores e o ano de publicação.

Tabela 6 – Artigos selecionados

| Base | Autores | Ano |
|----------------|--|------|
| Scopus | (GANDHI; THAKKER; JHA, 2017) | 2017 |
| Scopus | (PRADEEPKANDHASAMY; PRIYA; CHELLAPPAN, 2021) | 2021 |
| Science Direct | (CALANCEA et al., 2019) | 2019 |
| Science Direct | (MARTILLANO et al., 2019) | 2019 |

Fonte: Autores (2021)

Além dos trabalhos adicionados nas pesquisas retornadas através da *string* de busca aplicadas nas bases, foram adicionados mais 6 artigos, [Tabela 7](#), devido à relevância do conteúdo, as definições para a [Capítulo 2](#) e a aplicabilidade neste estudo.

Tabela 7 – Estudos Adicionados

| Estudos | Abordagem |
|----------------------------------|--|
| Coelho, Soares e Roehrs (2019) | Educação especial: Pcn x bncc |
| Torres e Ramirez (2019) | <i>Display Braille</i> |
| Possato e Monteiro (2020) | TIC |
| Santiago et al. (2020) | <i>Display Braille</i> |
| Shodiq, Pahlevi e Sukarno (2021) | Segurança MQTT |
| Ramos-García et al. (2022) | Arquitetura IoT e <i>Display Braille</i> |

Fonte: Autores 2023.

3.1.4 Resumos dos trabalhos

O trabalho de Pradeepkandhasamy, Priya e Chellappan (2021), busca a inclusão de deficientes visuais em fóruns de discussão social, nos quais a transmissão das informações ocorre por meio da fala. Para alcançar esse objetivo, os autores propõem a utilização de três componentes principais: Voz para Texto, Texto para Braille e Texto para Voz. Esses componentes permitem que o conteúdo seja convertido de fala para texto, de texto para Braille e de texto para fala novamente. Além disso, o sistema possibilita salvar, ouvir e deletar as mensagens recebidas.

Com a arquitetura apresentada, os arquivos armazenados no sistema podem ser facilmente convertidos para o formato Braille e posteriormente impressos, o que amplia ainda mais a acessibilidade para os usuários com deficiência visual. Esse trabalho representa uma importante contribuição para tornar os fóruns de discussão mais inclusivos e acessíveis para todas as pessoas, independentemente de suas habilidades visuais.

O artigo de [Calancea et al. \(2019\)](#), relatam da dificuldade que os deficientes visuais possuem em interagir com tecnologias devido à troca e processamento de informações serem basicamente por meio de textos escritos, imagens ou vídeos. Busca a viabilidade de recursos em nuvem, confiável, de baixa latência e flexível devido à arquitetura sem servidor. O projeto de aplicativo móvel, voltado para interagir com um sistema baseado em nuvem, traz uma abordagem de processamento distribuído, garantindo ao usuário resposta de seus assistentes pessoais e com a possibilidade de retorno em tempo real.

O estudo realizado por [Gandhi, Thakker e Jha \(2017\)](#), aborda as tecnologias disponíveis que auxiliam deficientes visuais, abrangendo desde telefones inteligentes até leitoras e aplicativos que permitem a digitação em Braille sensível ao toque. Entretanto, os autores destacam que, apesar das inovações existentes, ainda há tutores que utilizam tábuas com pontos ou recuos para ensinar o código Braille individualmente a cada aluno. Esse método tradicional demanda mais tempo e requer treinamento especializado dos tutores. Com o intuito de superar esses desafios, [Gandhi, Thakker e Jha \(2017\)](#), apresentam uma arquitetura de ensino em massa do código Braille, tornando o processo educacional mais flexível, rápido e acessível, sem complexidade de utilização. Eles propõem o uso de um microcontrolador baseado em atuador e Cortex-M4, implementado no projeto “Braille Cell”, que possibilita que um tutor ensine simultaneamente vários estudantes.

Quanto ao estudo de [Martillano et al. \(2018\)](#), ele teve como objetivo o desenvolvimento de uma tecnologia assistiva portátil e de baixo custo, capaz de auxiliar deficientes visuais e professores na prática e aprendizagem da escrita em notação Braille. O dispositivo chamado “*Braille Notation Device*” foi projetado para facilitar o aprendizado do código Braille e torná-lo mais acessível aos usuários.

3.1.5 Discussões a respeito dos trabalhos

Para um melhor entendimento, classificamos as subseções que direcionam os dados extraídos de cada estudo selecionado e seus respectivos autores.

3.1.5.1 Tecnologias utilizadas

Os estudos selecionados apresentam diversas tecnologias utilizadas para a implementação dos sistemas:

No trabalho de [Salvi, Pahar e Kadale \(2021\)](#), o sistema *Smart Glass* emprega uma série de tecnologias para torná-lo acessível a pessoas surdas e mudas, bem como a deficientes visuais. São utilizados painéis OLED transparentes (*T-Oled*) para visualização de conversões de voz em texto no *Universal Head-Up Display* (HUD) dos óculos. O sistema também pode empregar painéis OLED normais e um espelho para refletir o texto na frente do olho do usuário, evitando a necessidade de adicionar mais *hardware* pendurado. Câmeras IP são usadas para enviar a filmagem bruta para outros dispositivos móveis, onde é processada e convertida para reprodução.

Sensores de proximidade e ultrassônicos são empregados para detectar obstáculos e calcular distâncias. A pilha de tecnologia inclui ainda um *microphone array* para receber e processar áudio conforme o local do usuário, permitindo a detecção de veículos e outros eventos.

No estudo de Pradeepkandhasamy, Priya e Chellappan (2021), o sistema foi implementado usando linguagens de *script* do lado do cliente, como *Python Rest API* e *MongoDB*. Para reconhecimento de voz, foram utilizados *Web Kit Speech Recognition Pattern* e *Annyang.js*. Para a conversão de texto para Braille e vice-versa, foram empregadas tecnologias baseadas em *CSS* e *JQuery*. Essa abordagem é considerada leve e de código aberto, tornando o sistema acessível e colaborativo.

O projeto de (GANDHI; THAKKER; JHA, 2017), chamado *Braille Cell*, utiliza uma série de tecnologias para o ensino em massa do código Braille. O sistema é composto por um microcontrolador *STM32F429IDISCOVERY* baseado em *Cortex M4*, que permite a interação do tutor com os alunos. A comunicação sem fio é realizada utilizando o módulo *NRF24L01+*, que opera em 2,4 GHz. O sistema tátil, chamado de *Braille Cell*, é um dispositivo que permite aos alunos sentir o código Braille para cada letra transmitida pelo tutor. O atuador Mecanismo para o *Braille Cell* utiliza relés sem invólucros dispostos em camadas para levantar e abaixar os pinos que representam os ponto Braille. Essa configuração permite que vários alunos recebam o ensinamento simultaneamente.

Calancea et al. (2019) descreve o uso de tecnologias em nuvem e arquitetura sem servidor para a viabilização de recursos confiáveis, de baixa latência e flexíveis. O projeto utiliza aplicativo móvel e processamento distribuído para fornecer respostas em tempo real aos assistentes pessoais dos usuários, buscando atender às necessidades dos deficientes visuais.

3.1.5.2 Braille e interação tátil

Dentre os estudos selecionados, o código Braille e a interação tátil são temas centrais. O trabalho de Pradeepkandhasamy, Priya e Chellappan (2021), busca a inclusão de deficientes visuais em fóruns de discussão social por meio da conversão de fala para texto e, posteriormente, para Braille. O sistema utiliza uma interface tátil que permite aos usuários com deficiência visual sentir o conteúdo convertido em Braille. Isso amplia significativamente a acessibilidade e a interação dos deficientes visuais com o ambiente digital.

Gandhi, Thakker e Jha (2017) por sua vez, concentra-se no ensino do código Braille para deficientes visuais, especialmente no contexto educacional. O sistema *Braille Cell* utiliza um dispositivo tátil que permite aos alunos sentirem o código Braille para cada letra transmitida pelo tutor. A abordagem inovadora do ensino em massa do código Braille torna o processo educacional mais eficiente e acessível, permitindo que vários alunos sejam ensinados simultaneamente.

3.1.5.3 Assistência e comunicação

Além da interação tátil e do código Braille, os estudos selecionados também abordam a assistência e a comunicação com dispositivos externos. O *Smart Glass* desenvolvido por Salvi, Pahar e Kadale (2021), utiliza câmeras IP e um *microphone array* para auxiliar deficientes visuais e auditivos. O sistema permite a detecção de obstáculos e a conversão de voz em texto, fornecendo informações úteis aos usuários com deficiência.

Calancea et al. (2019) propõe um sistema baseado em nuvem e com arquitetura sem

servidor para permitir uma comunicação eficiente entre os assistentes pessoais dos usuários e o aplicativo móvel. Essa abordagem garante uma resposta rápida e em tempo real, atendendo às necessidades dos deficientes visuais.

Em resumo, os trabalhos selecionados apresentam uma variedade de tecnologias e abordagens para tornar a tecnologia mais acessível e inclusiva para deficientes visuais. O código Braille e a interação tátil são aspectos fundamentais para ampliar a acessibilidade e a interação dos usuários com o ambiente digital, enquanto a assistência e a comunicação eficiente com dispositivos externos aprimoram a experiência e a independência dos deficientes visuais.

3.1.5.4 Arquiteturas e aplicabilidade

O trabalho de [Salvi, Pahar e Kadale \(2021\)](#), descreve a arquitetura e a aplicabilidade do produto *Smart Glass* para auxiliar pessoas com dificuldades na visão. O sistema utiliza um ESP32 como placa de desenvolvimento, permitindo o envio de códigos diretamente para ele e o envio de vídeos para o servidor. A câmera IP é acessada em Python, utilizando a biblioteca OpenCV para o processamento de vídeo e detecção de objetos. O sistema consegue identificar objetos em tempo real e, caso haja um obstáculo detectado, converte o texto do objeto detectado em fala e o envia para um módulo de alto-falante conectado, notificando o usuário sobre o obstáculo à frente ou atrás.

Além disso, o sistema também utiliza sensores de proximidade para alertar o usuário sobre obstáculos laterais e permitir a escolha de uma nova rota. Para auxiliar pessoas surdas e mudas, o sistema possui microfones conectados para coletar sons do ambiente e os exibe no *HUD* do *Smart Glass* como mensagens de texto. O sistema também realiza a conversão de voz em texto e a reprodução desses textos no alto-falante conectado ao *Smart Glass*.

O estudo de [Pradeepkandhasamy, Priya e Chellappan \(2021\)](#), apresenta a arquitetura do aplicativo *Voice o Braille*, direcionado aos tutores ou apresentadores que auxiliam deficientes visuais em fóruns de discussão social. O aplicativo permite o reconhecimento de fala e sua conversão em texto, exibido na tela. Os usuários podem salvar, ouvir, revisar ou converter o texto em Braille. Caso o texto seja convertido em Braille, o usuário pode imprimi-lo. A arquitetura envolve a interação do usuário com o aplicativo móvel, que processa a fala e exibe o texto na tela. Posteriormente, é possível converter esse texto para o sistema Braille e imprimi-lo para acesso tátil.

O projeto de [Gandhi, Thakker e Jha \(2017\)](#), visa ensinar o código Braille em massa, utilizando um dispositivo Braille que possui modos de tutorial e *Drill*. No modo tutorial, os alunos iniciantes aprendem a posição dos pontos em uma célula Braille por meio de instruções de áudio e feedback tátil ao pressionar botões correspondentes aos pontos. No modo *Drill*, os alunos recebem palavras de três caracteres e, por meio do sistema de áudio auxiliar, são instruídos a soletrar as palavras, pressionando os botões correspondentes aos pontos de cada caractere.

Em resumo, os estudos apresentam diferentes arquiteturas e aplicabilidades para tornar a tecnologia mais acessível e inclusiva para deficientes visuais. O *Smart Glass* auxilia na detecção e notificação de obstáculos em tempo real, enquanto o aplicativo *Voice o Braille* permite a conversão de fala para texto e texto para Braille. O dispositivo Braille com modos de tutorial e *Drill* visa ensinar o código Braille de forma mais eficiente para vários alunos simultaneamente. Cada abordagem possui seus pontos fortes e específicos para atender às necessidades dos deficientes visuais em diferentes contextos.

3.1.5.5 Benefícios relatados

Os estudos apresentam diversos benefícios relatados:

Salvi, Pahar e Kadale (2021) destaca os benefícios do *Smart Glass* para pessoas com dificuldades na visão, que incluem a detecção de objetos, possibilitando maior independência na leitura, gestão de dinheiro e interação com o ambiente, além de ajudar na navegação ao evitar obstáculos.

Pradeepkandhasamy, Priya e Chellappan (2021) aponta que o aplicativo Voice o Braille representa uma melhoria significativa em relação às ferramentas disponíveis atualmente, permitindo a conversão de voz para texto e texto para Braille. Isso possibilita uma melhor documentação e armazenamento do conteúdo falado, facilitando o acesso e a revisão posterior.

Gandhi, Thakker e Jha (2017) enfatiza que o sistema de ensino em massa utilizando o dispositivo Braille oferece maior eficiência no aprendizado do código Braille, possibilitando que um tutor ensine vários alunos simultaneamente.

Martillano et al. (2018) menciona os benefícios do dispositivo Braille desenvolvido, sendo de baixo custo e utiliza materiais facilmente disponíveis, tornando-se uma opção acessível para auxiliar alunos com deficiência visual. O aplicativo móvel também permite a criação e envio de exercícios para o dispositivo, facilitando o processo de aprendizado.

3.1.5.6 Dificuldades e desafios

As pesquisas também apontam algumas dificuldades e desafios:

Pradeepkandhasamy, Priya e Chellappan (2021) destaca a dificuldade no reconhecimento de voz e na conversão de voz para Braille, bem como o alto custo de impressoras Braille disponível no mercado.

Martillano et al. (2018) ressalta que o principal desafio do estudo foi inovar um suporte tecnológico de escrita de notação Braille básico no *SPED Center* eficientemente e de baixo custo. Além disso, os recursos e design limitados em tecnologias assistivas para alunos iniciantes em Braille foram considerados um desafio.

4

Análise e Arquitetura

Com base nos estudos de Pradeepkandhasamy, Priya e Chellappan (2021) e sua arquitetura Figura 10, a arquitetura deste trabalho visa fornecer uma visão orientada pelo seu projeto Voice o Braille, porém apresentando as seguintes alterações:

Dispositivo de saída: a arquitetura, altera quanto ao dispositivo de saída (output), *display* Braille, representado na Figura 11. A arquitetura Voice o Braille dispõe de uma impressora Braille como saída, encarecendo a arquitetura. Segue abaixo alguns valores das impressoras Braille, conforme pesquisa realizada em 08 de agosto 2023:

loja virtual **Mundo da Lupa**¹, conforme Tabela 8

Tabela 8 – Impressoras Braille e seus valores à vista

| Impressora | Descrição | Valor à vista (R\$) |
|------------------|------------------------------|---------------------|
| Braille MAX | Impressora Braille MAX | R\$37.150,00 |
| Columbia 2 | Impressora Braille Columbia | R\$28.997,00 |
| Delta | | R\$37.997,00 |
| Elite 200 | Elite 200 | R\$97.500,00 |
| Embraille | Impressora Embraille | R\$20.495,00 |
| Emprint Spot Dot | Impressora Emprunte SPOT DOT | R\$44.995,00 |
| Premier 100 | Premier 100 | R\$66.995,00 |
| Rogue Folha | Impressora Rogue frontal | R\$56.997,00 |
| Rogue Trator | Impressora Rogue frontal | R\$49.997,00 |

Fonte: <<https://mundodalupa.com.br/categoria-produto/braille/impressora-braille/>>

Segundo a loja virtual **ViewPlus**², conforme Tabela 9:

¹ <<https://mundodalupa.com.br/categoria-produto/braille/impressora-braille/>>

² <<https://viewplus.com/pt/product-category/braille-embossers/>>

Tabela 9 – Impressoras Braille e seus preços em reais

| Impressora | Faixa de Preço (US\$) | Faixa de Preço (R\$) |
|----------------------------|------------------------------|------------------------------|
| Columbia | \$3.495 — \$4.684 | R\$16.995,00 — R\$22.797,00 |
| Embraille | \$1.995 — \$2.674 | R\$9.721,00 — R\$13.053,00 |
| MAX substituída pela ROGUE | \$5,995 — \$8,704,00 | R\$29.195,00 — R\$42.358,00 |
| SPOTDOT | \$7.995 — \$10.714 | R\$38.955,00 — R\$52.089,00 |
| Rogue | \$5.995 — \$8.704 | R\$29.195,00 — R\$42.358,00 |
| Delta | \$4.195 — \$5.624 | R\$20.413,00 — R\$27.436,00 |
| Premier | \$9.995 — \$18.748 | R\$48.674,00 — R\$91.318,00 |
| Elite | \$14.995 — \$25.448 | R\$73.042,00 — R\$123.920,00 |

Fonte: <<https://viewplus.com/pt/product-category/braille-embossers/>>

De acordo com Torres e Ramirez (2019), o seu projeto de *display* Braille custou em média **R\$129,67**, conforme mostra Tabela 10, considerando o salário mínimo em 2018.

Tabela 10 – Lista de Materiais e Custos

| Material | Quantidade | Custo Unitário (R\$) | Custo Total (R\$) |
|--------------------------|-------------------|-----------------------------|--------------------------|
| Polímero ABS | 87 gramas | R\$119,00/kg | R\$10,43 |
| Atuador Solenoide | 6 un. | R\$4,42 | R\$26,52 |
| Arduino Uno + cabo USB | 1 un. | R\$39,90 | R\$39,90 |
| Diodo de proteção N 4002 | 6 un. | R\$0,07 | R\$0,42 |
| Resistor 100 OHM | 6 un. | R\$4,00 | R\$24,00 |
| Transistor PN3568 5V 1A | 6 un. | R\$4,10 | R\$24,60 |
| Botão Switch H | 2 un. | R\$1,00 | R\$2,00 |
| Mini Parafuso M1 x 4,5mm | 4 un. | R\$0,45 | R\$1,80 |

Fonte: Santiago et al. (2020)

Para Santiago et al. (2020), relata que, se já for disponibilizada uma oficina completa (representada pelos itens da Tabela 11 e Tabela 12), o custo de cada Celta ficará próximo ao total da Tabela 13, sendo o material para o circuito-elétrico. Os outros materiais de consumo serviriam para a montagem de diversos Celtas. De qualquer modo, se somarmos os 3 quadros e mais o valor de uma impressora 3D (de cerca de R\$2.000,00), uma instituição deveria gastar cerca de R\$3.500,00 com equipamentos e materiais para um Celta e depois mais cerca de R\$ 300,00 para fazer cada um dos Celtas seguintes.

Tabela 11 – Material Permanente para o Projeto

| Quantidade e Descrição do Item | Valor Aproximado (R\$) |
|--|-------------------------------|
| 1 Retífica Dremel 3000 com 10 Acessórios, 110v | R\$ 260,00 |
| 1 Fonte de bancada 0 - 15v 2A | R\$ 155,00 |
| 1 Jogo de chaves fenda/phillips | R\$ 35,00 |
| 1 Multímetro digital convencional | R\$ 45,00 |
| 1 Mini morsa (torno de bancada) | R\$ 30,00 |
| 1 Paquímetro digital | R\$ 65,00 |
| 1 Ferro de soldar | R\$ 45,00 |
| Valor total | R\$ 590,00 |

Fonte: [Santiago et al. \(2020\)](#)

Tabela 12 – Material de Consumo para a Bancada

| Quantidade e Descrição do Item | Valor Aproximado (R\$) |
|--|-------------------------------|
| 1 Kit brocas precisão 0,8 A 3,2 Mm Ref. 628 Com 7 Peças Dremel | R\$ 38,00 |
| 1 Kit ferramentas para retífica Dremel uso geral | R\$ 117,00 |
| 1 Pistola de cola quente + refil | R\$ 20,00 |
| 1 Jogo de chaves fenda/phillips | R\$ 35,00 |
| 1 Espaguete Termo Contrátil - 4 Bitolas (2, 3, 4 e 6mm x 5m) | R\$ 16,00 |
| 1 Kit de limas Agulha para acabamento | R\$ 43,00 |
| 2 <i>Protoboards</i> de 400 pontos | R\$ 20,00 |
| 1 Filamento ABS 1.75mm 1kg Branco | R\$ 85,00 |
| 1 Filamento ABS 1.75mm 1kg Preto | R\$ 85,00 |
| 2 Rolos de fita Isolante | R\$ 10,00 |
| 1 Rolo de Estanho para solda | R\$ 20,00 |
| 1 Rolo de fita crepe | R\$ 10,00 |
| 1 Kit para confecção de placa de circuito impresso | R\$ 160,00 |
| Valor total | R\$ 659,00 |

Fonte: [Santiago et al. \(2020\)](#)

Tabela 13 – Material de Consumo para o Circuito Elétrico e a Montagem Final

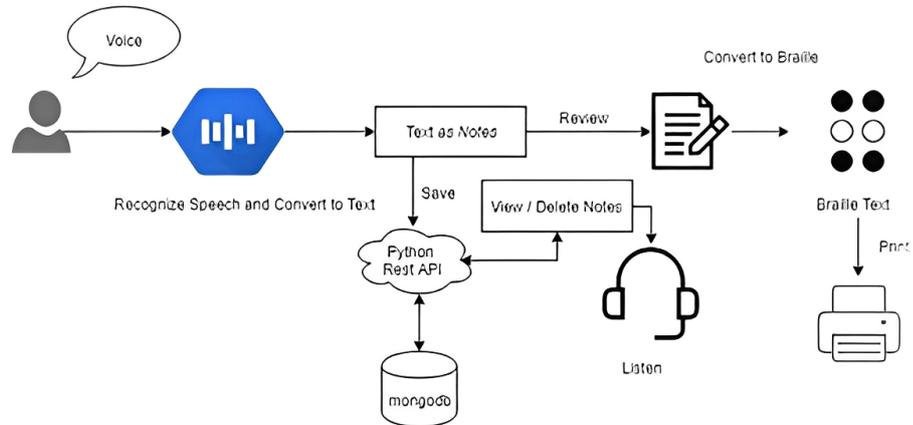
| Quantidade e Descrição do Item | Valor Aproximado (R\$) |
|---|-------------------------------|
| 3 Conectores DIP-1X2P-2.54MM | R\$ 14,00 |
| 2 Conectores DIP 2510S-12P | R\$ 10,00 |
| 3 Soquetes DIP-16 | R\$ 9,00 |
| 2 Resistores de 330 ohms, 5%, 1/4W | R\$ 3,00 |
| 6 Motores de <i>vibracall</i> de celular | R\$ 90,00 |
| 2 <i>Push bottons</i> com 12mm de diâmetro para encaixe | R\$ 6,00 |
| 3 CIs L293D | R\$ 39,00 |
| 1 Arduino Uno R3 | R\$ 30,00 |
| 3 Barras de 40 pinos Fêmea 180º | R\$ 9,00 |
| 1 Barra de soquete <i>header</i> | R\$ 1,00 |
| 5 Metros de cabos flexíveis 24 AWG, 0,20mm de cores variadas | R\$ 5,00 |
| 1 Pacote com 40 Cabos flexíveis com conectores <i>Dupont</i> macho e fêmea | R\$ 15,00 |
| 1 Placa ilhada especial para <i>Shield</i> de Arduino Uno R3 | R\$ 20,00 |
| 1 Placa folheada de cobre, dupla face para PCB | R\$ 10,00 |
| 2 Parafusos de 30mmX2mm com porca | R\$ 1,00 |
| 12 Parafusos de 7mmx2,5mm | R\$ 1,00 |
| 1 Fonte de alimentação DC 5V, 1A, positivo no centro e negativo por fora | R\$ 15,00 |
| Valor total | R\$ 278,00 |

Fonte: [Santiago et al. \(2020\)](#)

Arquitetura em nuvem: a arquitetura descrita no trabalho de pesquisa do Voice o Braille possibilita o reconhecimento de voz, a conversão do áudio em texto e, em seguida, a possibilidade de salvar, ouvir, revisar, deletar e até mesmo converter o texto em Braille para imprimir em uma impressora específica, que possibilita a saída da codificação em relevo no papel. No entanto, incorporar a Computação em Nuvem traz os seguintes benefícios:

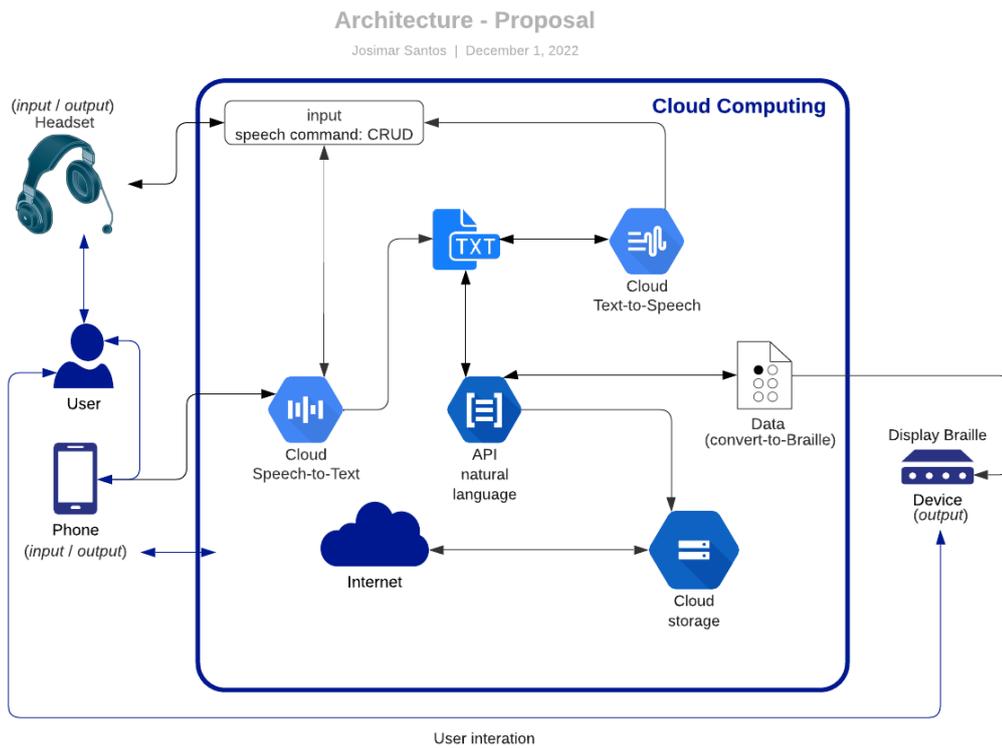
- Disponibilidade;
- Mobilidade;
- Flexibilidade;
- Segurança;
- Compartilhamento;
- Baixo custo em infraestrutura.

Figura 10 – Arquitetura Voice o Braille



Fonte: Pradeepkandhasamy, Priya e Chellappan (2021)

Figura 11 – Arquitetura



Fonte: Autores - 2022

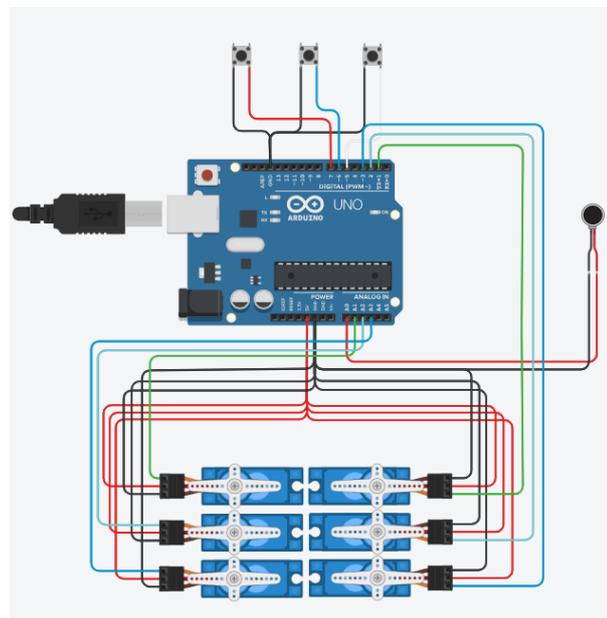
de: 2. Compete a arquitetura analisada, conforme Tabela 14, estrutura computacional capaz

Tabela 14 – Descrição e Resultados

| Descrição | Resultados |
|--|---------------------------|
| a. Reconhecer comandos de voz | Implementado com sucesso |
| b. Reconhecer caracteres por meio de um teclado | Implementado com sucesso. |
| c. Reconhecer caracteres por meio de um <i>display</i> de toque | Implementado com sucesso |
| d. Processar e converter os caracteres. fornecidos por meio da fala, teclado ou <i>display</i> de toque para notação Braille | Implementado com sucesso |
| e. Processar e converter os caracteres. fornecidos para notação Braille | Implementado com sucesso |
| f. Definir o processo de conversão conforme a modelagem da arquitetura | Implementado com sucesso |
| g. Possibilitar o CRUD (<i>Create, Read, Update, Delete</i>) de caracteres para composição de palavras | Implementado com sucesso |
| h. Desenvolver um modelo em simulador online | Implementado com sucesso. |

Fonte: Autores 2023

Figura 12 – Simulador tinkercad: Modelagem de componentes



Fonte: Autores (2022) cópia editada em (2023)

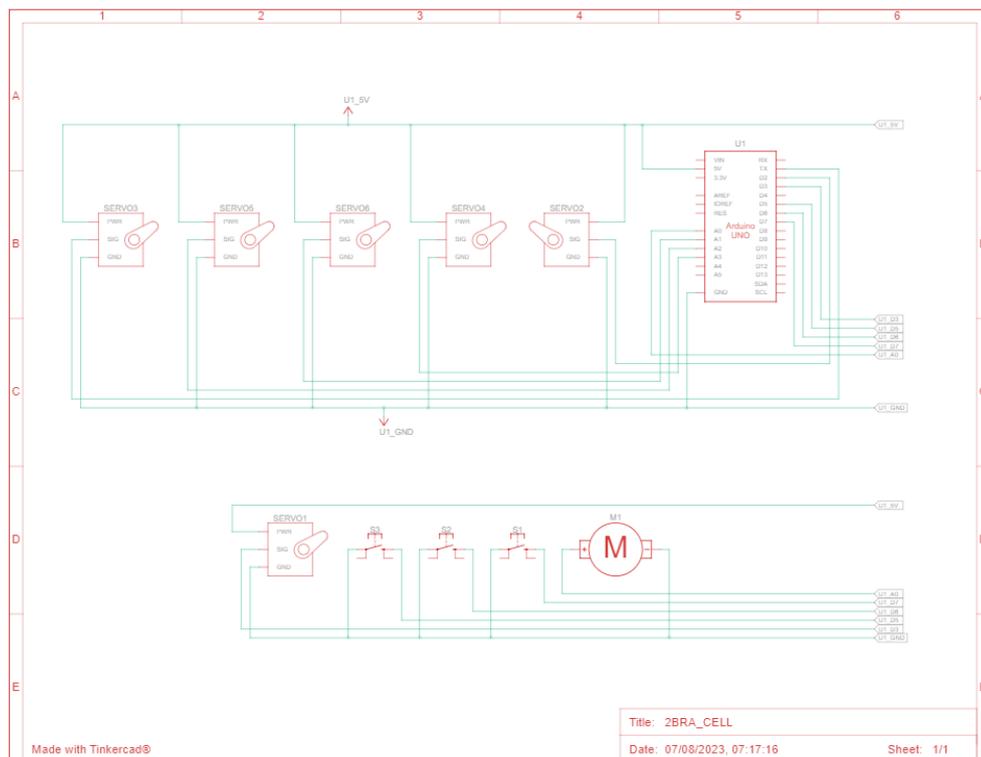
<<https://www.tinkercad.com/things/8bg6ycMMfY2-2bracell>>

Figura 13 – Simulador tinkercad: Lista de componentes do modelo de teste

| Nome | Quantidade | Componente |
|--|------------|------------------------|
| U1 | 1 | Arduino Uno R3 |
| SERVO2 SERVO4 SERVO6 SERVO5 SERVO3 SERVO1 | 6 | Posicional Micro servo |
| M1 | 1 | Motor de vibração |
| S1 S2 S3 | 3 | Botão |

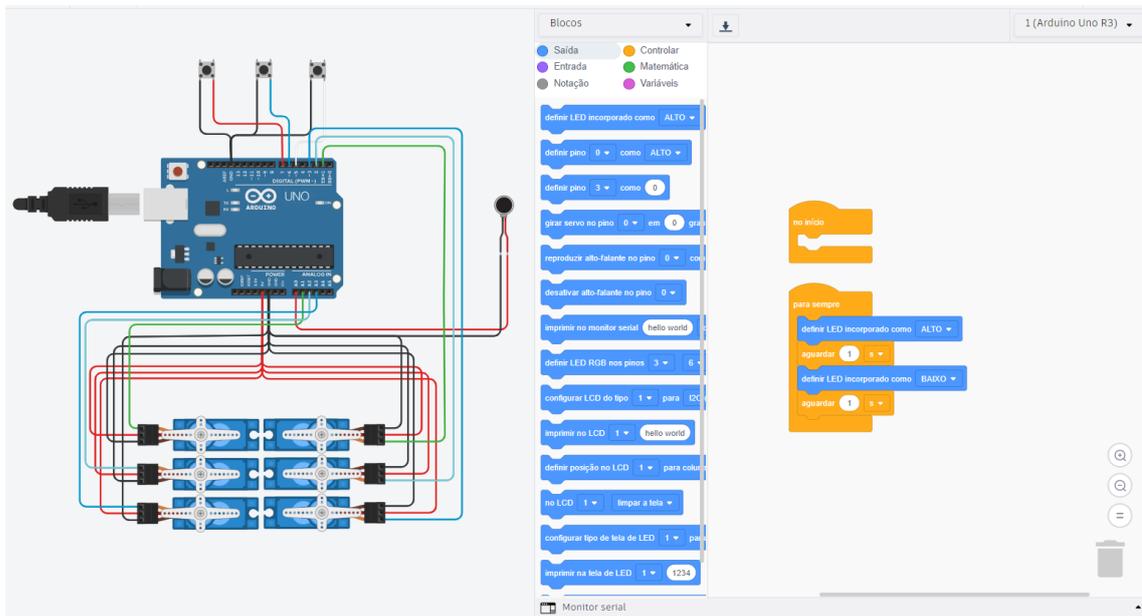
Fonte: Autores (2022) cópia editada em (2023)
 <<https://www.tinkercad.com/things/8bg6ycMMfY2-2bracell>>

Figura 14 – Simulador tinkercad: Visão esquemática



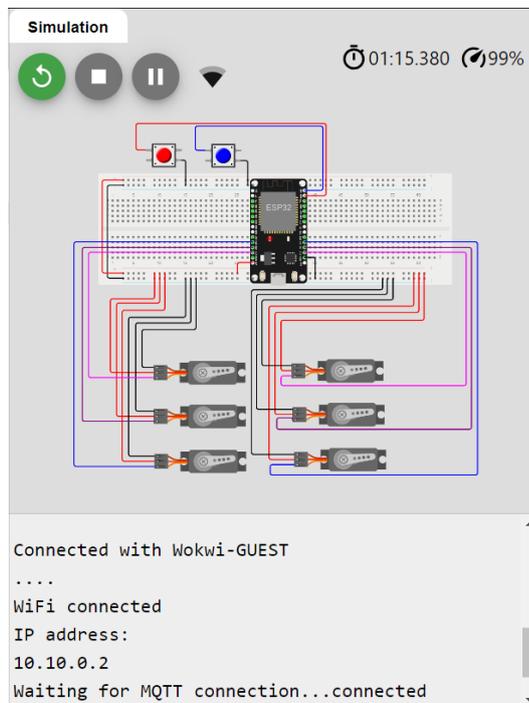
Fonte: Autores (2022) cópia editada em (2023)
 <<https://www.tinkercad.com/things/8bg6ycMMfY2-2bracell>>

Figura 15 – Simulador tinkercad: Codificação por blocos



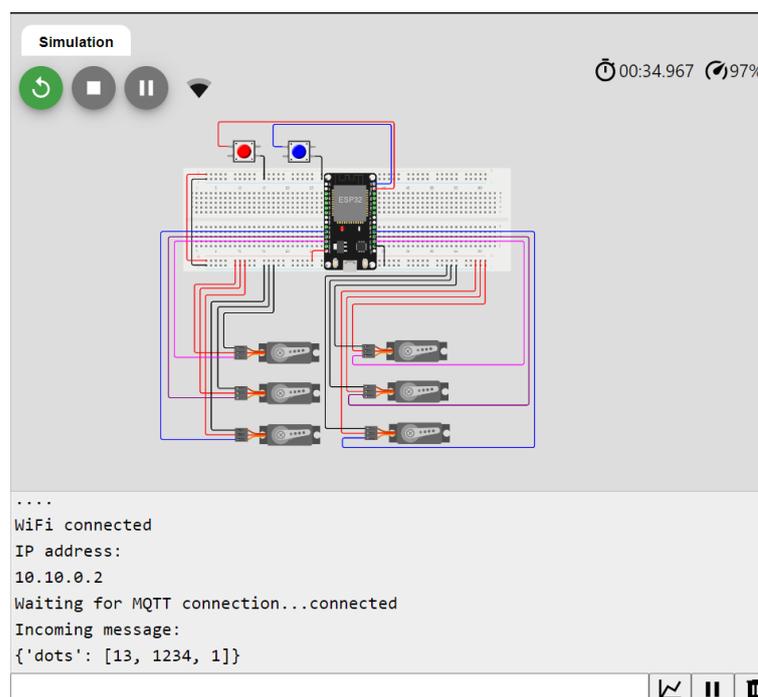
Fonte: Autores (2022) cópia editada em (2023)
 <<https://www.tinkercad.com/things/8bg6ycMMfY2-2bracell>>

Figura 16 – Simulador wokwi: Primeira conexão via MQTT



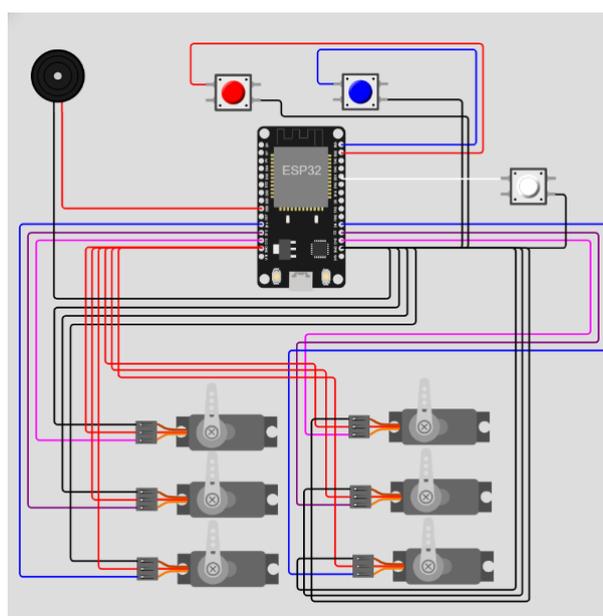
Fonte: Autores (2022) cópia editada em (2023)
 <<https://wokwi.com/projects/348875560981627476>>

Figura 17 – Simulador wokwi: Recepção da mensagem após assinar no broker



Fonte: Autores (2022) cópia editada em (2023)
<<https://wokwi.com/projects/348875560981627476>>

Figura 18 – Simulador wokwi: Arduino ESP32 versão atual



Fonte: Autores (2022) cópia editada em (2023)
<<https://wokwi.com/projects/348875560981627476>>

1. Exploramos simuladores online com a possibilidade de realizar testes para criar um modelo de arquitetura com a mesma ideia apresentada na [Figura 11](#).

Simuladores testados para *display* Braille:

1. tinkercad ³
2. wokwi ⁴

Para o simulador 1 - Tinkercad: Um modelo foi desenvolvido, conforme [Figura 12](#), porém ocorreram algumas limitações, por exemplo, ausência de componentes e disponibilidade de conexão via *Internet*. Contudo, algumas disponibilidades como gerar lista de componentes, [Figura 13](#), gerar automaticamente esquemas clássicos de diagramas elétricos, [Figura 14](#), bem como dispõe da modelagem e execução de código por blocos, [Figura 15](#), facilitando a implementação lógica.

No simulador 2 - Wokwi: não encontramos problemas na utilização, conseguimos simular o Esp32, placa Arduino que já dispõe de configuração para acesso à rede e obtivemos conexão com êxito [Figura 16](#), entretanto em termos de gerar documentação o wokwi é limitado. Outro aspecto importante a ressaltar é que, no estágio inicial do desenvolvimento do modelo, optamos por usar uma *Protoboard*, no entanto, essa inclusão gerou problemas na execução individual dos servos motores. Posteriormente, ao conectar todos os componentes diretamente na placa do Arduino Nano, os servos motores passaram a funcionar de maneira esperada, como podemos comparar na [Figura 16](#) e [Figura 17](#) versão atual em [Figura 18](#).

4.0.1 Arquiteturas e aplicabilidade

O aplicativo desenvolvido por [Pradeepkandhasamy, Priya e Chellappan \(2021\)](#) possui um foco direcionado aos tutores e apresentadores que auxiliam deficientes visuais em seus fóruns de discussão. Esse sistema oferece diversas funcionalidades essenciais, incluindo o reconhecimento de fala e sua conversão em texto, exibido na tela. Uma vez exibido, o usuário tem a opção de salvar, ouvir ou revisar o texto, além da possibilidade de convertê-lo para o formato Braille.

Ao realizar a conversão para Braille, o usuário tem a vantagem de poder imprimir o texto, como ilustrado na [Figura 10](#) e [Figura 11](#), que demonstram o processo de reconhecimento de voz e a respectiva conversão em Braille.

Essa gama de funcionalidades torna o aplicativo uma ferramenta poderosa para o ensino e a inclusão de deficientes visuais, permitindo que eles participem de fóruns de discussão social com maior autonomia e acesso à informação. O aplicativo facilita a interação entre tutores e estudantes, bem como a produção e compartilhamento de conteúdo de maneira acessível.

Essa iniciativa mostra como a tecnologia pode ser empregada significativamente para criar soluções inovadoras e inclusivas, superando barreiras e tornando o ambiente educacional e social mais acessível a todos os indivíduos, independentemente de suas habilidades visuais. A convergência entre tecnologia e educação, como evidenciada neste aplicativo, abre novas possibilidades e oportunidades para melhorar a qualidade de vida e o desenvolvimento educacional de pessoas com deficiência visual.

³ <<https://www.tinkercad.com>>

⁴ <<https://wokwi.com>>

O estudo de [Gandhi, Thakker e Jha \(2017\)](#) aborda a questão do ensino do Código Braille para deficientes visuais. Atualmente, o método tradicional de ensino envolve o tutor segurando a mão do aluno e guiando-o para tocar os pontos em relevo da matriz Braille. Esse processo requer que o tutor seja treinado especificamente para o ensino desse código, e cada aluno precisa ser ensinado individualmente, consumindo uma quantidade significativa de tempo.

Para otimizar esse processo de ensino, [Gandhi, Thakker e Jha \(2017\)](#) propõem um conceito de sistema de ensino em massa, utilizando recursos sem fio para evitar o uso de cabos e permitir que os alunos não fiquem restritos a uma posição específica ou distância. Nesse sistema, o tutor pode digitar os caracteres em um teclado de tela sensível ao toque, e esses caracteres são transmitidos sem fio para o painel de cada aluno.

Essa abordagem possibilita que o tutor ensine vários alunos ao mesmo tempo, tornando o processo de aprendizado mais rápido e flexível. Além disso, a utilização de tecnologia sem fio oferece maior mobilidade aos alunos, permitindo que eles se posicionem de forma mais confortável durante as atividades de aprendizado.

Esse projeto demonstra como a tecnologia pode ser aplicada inovadoramente para melhorar o processo de ensino e aprendizagem do Código Braille, tornando-o mais acessível, eficiente e inclusivo. Ao promover a aprendizagem em massa, esse sistema possibilita que mais alunos com deficiência visual tenha acesso ao ensino do Braille de maneira mais ágil e eficaz.

A arquitetura apresentada por [Pradeepkandhasamy, Priya e Chellappan \(2021\)](#), oferece a funcionalidade de armazenamento das notações Braille geradas a partir da conversão de voz para texto. No entanto, é importante destacar que a obra não menciona a possibilidade de compartilhamento dessas notações com outras pessoas. Essa funcionalidade poderia ser interessante para facilitar a colaboração e o compartilhamento de informações entre diferentes usuários.

Outro aspecto relevante é que, embora o aplicativo permita o armazenamento das notações Braille e a audição em áudio do texto salvo, não é mencionada a possibilidade de edição ou correção dessas notações. Caso haja algum erro ou necessidade de ajuste no texto armazenado, a única opção disponível parece ser a exclusão do registro. A inclusão de uma função de edição ou correção poderia aumentar a usabilidade e a praticidade do aplicativo.

Uma alternativa para o armazenamento das notações Braille, que pode ser explorada em complemento ao estudo de [Pradeepkandhasamy, Priya e Chellappan \(2021\)](#), é a utilização de um banco de dados em nuvem, como o MongoDB, como mencionado por [\(CALANCEA et al., 2019\)](#). O armazenamento em nuvem oferece vantagens como baixa latência e flexibilidade, tornando o acesso e a recuperação das informações mais rápidos e facilitados. Além disso, uma arquitetura sem servidor pode ser implementada para otimizar o gerenciamento dos dados e garantir uma melhor escalabilidade do sistema.

Dessa forma, ao incorporar a funcionalidade de compartilhamento, permitir a edição ou correção das notações Braille armazenadas e explorar o armazenamento em nuvem com arquitetura sem servidor, o aplicativo poderia oferecer uma experiência mais completa, eficiente e abrangente para os usuários, contribuindo para uma melhor inclusão e acessibilidade de pessoas com deficiência visual.

4.1 Arquitetura

A arquitetura no projeto Voice o Braille apresenta diferenciais significativos em relação aos estudos de [Pradeepkandhasamy, Priya e Chellappan \(2021\)](#). Algumas das principais melhorias são:

Output: O desenvolvimento de um *display* Braille utilizando a ideia de [Gandhi, Thakker e Jha \(2017\)](#) para leitura dos caracteres em Arduino como terminal de saída proporciona mais economia quando comparado com a impressora Braille mencionada na arquitetura original ([PRADEEPKANDHASAMY; PRIYA; CHELLAPPAN, 2021](#)). A utilização de uma impressora 3D para confeccionar a caixa protetora e os pontos táteis do dispositivo permite maior praticidade e personalização.

Nuvem: A utilização de brokers para publicação/assinatura para o protocolo MQTT, juntamente com ferramentas de armazenamento e execução para a API de conversão na nuvem, texto para voz, voz para texto, oferece maior escalabilidade e flexibilidade para o sistema. Isso possibilita que o projeto seja acessado e utilizado em diferentes dispositivos, sem a necessidade de configurações complexas.

A arquitetura do projeto fornecerá, por meio de uma aplicação, instruções por comando de voz. Uma “assistente” irá falar o processo e solicitará comandos para a tomada de decisão, onde o usuário poderá responder com o comando desejado para dar continuidade ao fluxo de interação.

A conversão de voz para texto será realizada através do módulo SpeechRecognition [Anthony Zhang \(Uberi\) \(2014\)](#), versão 3.8.1. A biblioteca gTTS [Pierre Nicolas Durette and Contributors \(2014\)](#), versão 2.3.0, será utilizada para executar a conversão de texto para fala, com o auxílio da biblioteca PyAudio [Hubert Pham \(2006\)](#), versão 0.2.12. O projeto contará com uma API em linguagem Python [Python Core Team \(2019\)](#), versão 3.9, que gerenciará o arquivo de texto com extensão “.txt”. O usuário poderá retomar o arquivo posteriormente para continuar, solicitar que a assistente “leia” o texto contido no arquivo, editar partes do texto ou excluí-lo. Além disso, a API permitirá, através do comando “converter” [Figura 20](#), realizar a conversão do texto em um formato interpretado pelo dispositivo do *display* Braille.

Para a estrutura do *display* Braille, será utilizado o Arduino, especificamente um ESP32, servomotores e *pushbuttons* para simulação dos pontos em relevo. O *display* “exibirá” apenas um caractere por vez, sendo que o controle de passagem para o próximo caractere poderá ser gerenciado por botões no próprio *display* Braille.

A plataforma Tinkercad ⁵ foi utilizada para testar a simulação do dispositivo, no entanto, não houve disponibilidade para o ESP32 ou funcionalidades de simulação com conexão para troca de informações via *Internet* por motivos de segurança.

Com todos esses diferenciais, a arquitetura do projeto Voice o Braille se mostra como uma solução inovadora, acessível e inclusiva, contribuindo para a transformação da inclusão de deficientes visuais em fóruns de discussão social, proporcionando uma experiência de aprendizado mais eficiente e eficaz ([SANTIAGO et al., 2020](#)).

⁵ <https://www.tinkercad.com/dashboard>

5

Resultados e Discussão

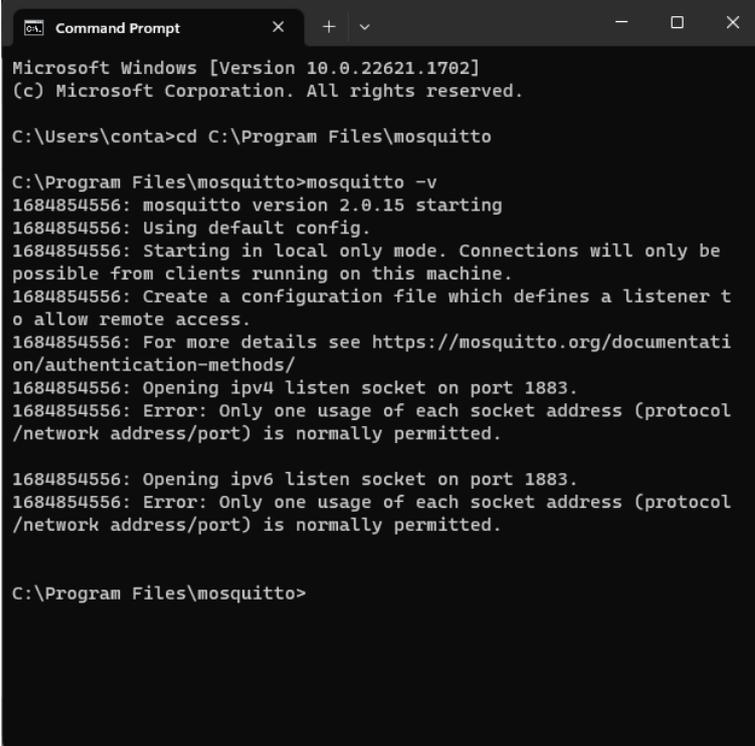
Este capítulo apresenta os resultados preliminares da arquitetura em plataformas de simulação, que estão atualmente em fase experimental.

5.1 Detalhamento dos Experimentos

Os resultados alcançados por meio da plataforma Wokwi durante os testes preliminares revelaram um grau satisfatório de desempenho. No entanto, durante essa fase, surgiram diversos desafios que se relacionaram, na maioria, com os atrasos inerentes à utilização das bibliotecas de conversão entre fala e texto, bem como de texto para fala. Um notável exemplo foi o retardo observado quando a "assistente" proferia o áudio pela primeira vez. Essa questão específica foi superada ao armazenar os arquivos de áudio após o primeiro acionamento, o que possibilitou a sua reutilização. É válido ressaltar que tal abordagem implicou a necessidade de disponibilizar espaço de armazenamento para esses arquivos.

Adicionalmente, foi identificado um conjunto de desafios relacionados à captação da fala do usuário, mesmo com a aplicação das técnicas de redução de ruído incorporadas na biblioteca. Como resultado, algumas palavras não puderam ser devidamente identificadas ou a captura em si não obteve êxito, o que potencialmente influenciou a experiência do usuário. Com o intuito de assegurar uma interação mais eficaz e otimizada, foram requeridos ajustes substanciais e iterações de teste para aprimorar esse aspecto.

Figura 19 – Mosquitto 2.0.15 - versão utilizada



```
Microsoft Windows [Version 10.0.22621.1702]
(c) Microsoft Corporation. All rights reserved.

C:\Users\conta>cd C:\Program Files\mosquitto

C:\Program Files\mosquitto>mosquitto -v
1684854556: mosquitto version 2.0.15 starting
1684854556: Using default config.
1684854556: Starting in local only mode. Connections will only be
possible from clients running on this machine.
1684854556: Create a configuration file which defines a listener t
o allow remote access.
1684854556: For more details see https://mosquitto.org/documentati
on/authentication-methods/
1684854556: Opening ipv4 listen socket on port 1883.
1684854556: Error: Only one usage of each socket address (protocol
/network address/port) is normally permitted.

1684854556: Opening ipv6 listen socket on port 1883.
1684854556: Error: Only one usage of each socket address (protocol
/network address/port) is normally permitted.

C:\Program Files\mosquitto>
```

Fonte: <<https://mosquitto.org/>>
Autores (2022)

Para a comunicação entre o servidor e o *display* Braille, a plataforma Wokwi, link para o projeto em desenvolvimento ¹, permitiu a simulação local e o uso do *broker* Mosquitto² Figura 19 para a comunicação via MQTT. No entanto, foi identificado um problema relacionado ao tamanho da mensagem em MQTT, uma vez que a API preparou um JSON Figura 21 e Figura 20, que compôs uma estrutura de dados, com o texto do arquivo solicitado pelo usuário, cujo tamanho pôde exceder o limite suportado pelo MQTT. Para solucionar essa questão, foi prevista a limitação da quantidade de caracteres armazenados em cada arquivo de texto, garantindo que as mensagens ficassem no limite suportado.

¹ <https://wokwi.com/projects/348875560981627476>

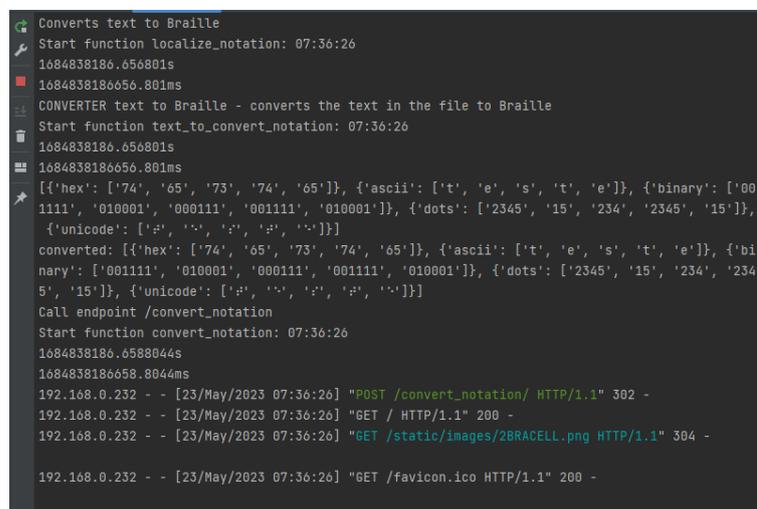
² <https://mosquitto.org/>

Figura 20 – API: Tela com entrada de texto para conversão à esquerda e tela com texto "teste", convertido à direita.



Fonte: Autores (2022)

Figura 21 – API: Envio do JSON por mensagem MQTT para ESP32 simulador



Fonte: Autores (2022)

A palavra ou frase foi substituída na conversão Braille por uma estrutura que permitiria futuramente ser trabalhada em diversos aspectos. A base de dados na qual as letras, números e caracteres foram convertidos através das suas respectivas representações fornecidas por um arquivo `ssbraille.csv`³, disponível na plataforma do kaggle⁴. O arquivo foi atentamente

³ <https://www.kaggle.com/datasets/josimarsts/ssbraille?select=ssbraille.csv>

⁴ <https://www.kaggle.com/>

atualizado, que atendeu as necessidades de conversão Braille, como, por exemplo, iniciadas com letras maiúsculas, bem como palavras para a escrita português do Brasil, exemplo c e ç. Um exemplo abaixo, da palavra “test” mostrou a estrutura e tipos de conversão, enviadas pela API para o *display* via MQTT. O hexadecimal, ou simplesmente **hex**, representou os caracteres em hexadecimal conforme tabela ASCII ⁵. O **ascii**, conforme tabela ASCII ⁶. O binário, **binary**, enviou um conjunto de 6 bits entre 0 e 1 para cada carácter da palavra, que pôde ser representado por 0 (zero), quando o ponto estava baixo e 1 quando estava alto. O **dots**, representou um conjunto de 6 números de 1 a 6 que indicou qual ponto seria elevado, por exemplo, se no conjunto compusesse “12” o ponto 1 e 2 foi erguido, já os números “345”, ausentes do conjunto, representaram os pontos baixos (SANTIAGO et al., 2020). E o **unicode** que representou cada carácter em seus respectivos pontos em Braille utilizado para exibição. Em se tratando de exibição, o artigo direcionou-se para solução de arquitetura que possibilitassem a beneficiar pessoas com deficiência visual, bem como professores e interessados. Então, além de possibilitar a comunicação com a API por comando de voz, a arquitetura disponibilizou uma interface para conversão. O projeto de pesquisa visou a integração da arquitetura Cloud e do protocolo MQTT com dispositivos IoT Arduino, juntamente com a utilização de Braille para exibição de informações. O foco principal não esteve na comercialização dos resultados, mas sim em promover a acessibilidade e inclusão de pessoas com deficiência visual no contexto da *Internet das Coisas*. O propósito foi desenvolver uma solução tecnológica que permitisse que pessoas com deficiência visual pudessem interagir de forma mais independente com dispositivos e serviços IoT, fornecendo-lhes acesso a informações relevantes e melhorando sua qualidade de vida. A pesquisa visou contribuir para o avanço do conhecimento e ofertar uma aplicação prática que beneficiasse a sociedade, especialmente aqueles com necessidades especiais de acessibilidade. Estes aspectos de conversão podem ser observados nas figuras: [Figura 21](#) e [Figura 20](#)

Código 2 – Código *Publish/Subscribe*

```
1 client.publish("/twobrancell", '{"dots': [13, 1234, 1]}");  
2 client.subscribe("/twobrancell");
```

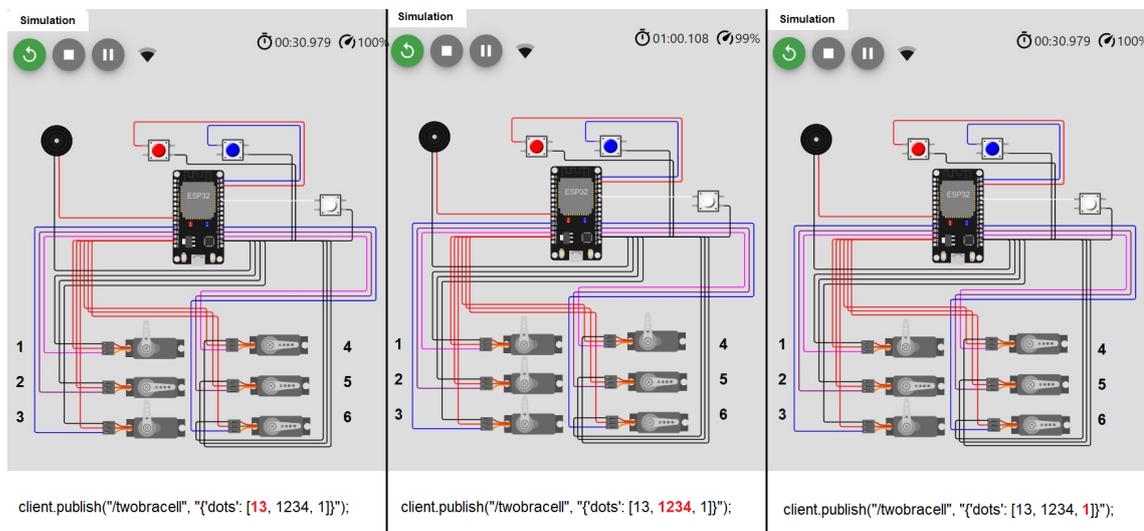
Fonte: Autores (2023)

É importante ressaltar que o MQTT é um protocolo de mensagens leves frequentemente utilizado em sistemas de comunicação entre dispositivos, sendo especialmente relevante em cenários de *Internet das Coisas* (IoT). A configuração e uso do broker e/ou servidor, tópicos e mensagens são fundamentais para garantir uma comunicação eficaz e confiável entre os dispositivos (TORRES; RAMIREZ, 2019).

⁵ <https://www.ascii-code.com/>

⁶ <https://www.ascii-code.com/>

Figura 22 – Simulador wokwi: Arduino ESP32 em execução



Fonte: Autores (2023)

<<https://wokwi.com/projects/348875560981627476>>

No ambiente do Wokwi, a conexão foi estabelecida utilizando o endereço IP 10.10.0.2, direcionando a comunicação para o servidor broker hospedado em test.mosquitto.org na porta 1883, podemos utilizar como teste gratuitamente, também no broker.hivemq.com e mesma porta 1883, servindo como alternativa em caso de falhas de comunicação com o test.mosquitto.org. O tópico de comunicação escolhido foi “/twobrancell”.

Na Figura 22, a linha 2, ilustra o trecho de código responsável por publicar um JSON de teste. Esse JSON representa, após conversão em caracteres, a sequência de letras “[kpa]”, que foi inserida de maneira aleatória apenas para fins de teste. Dois botões, componente *pushbutton*⁷ foram adaptados ao protótipo com intuito de percorrer a “palavra” ou “frase” convertida.

O botão **azul**, conforme mostra a Figura 22, compete a função de seguir um passo a frente da array de caracteres o primeiro quadro a esquerda mostra como os servomotores [1,3] se comportaram ao acionar o caractere [k] ou “dots”:[13,1234,1], ao pressionar novamente o botão novamente, segue para o próximo caractere [p] ou, diante da conversão, “dots”:[13,1234,1], quadro ao centro da Figura 22, servomotores [1234] se movem ao angulo destinado, e por fim novamente pressionado chegamos ao fim da lista, no caractere [a] ou “dots”:[13,1234,1], em que apenas o servomotor [1], Figura 22, quadro a direita, se move para representar a letra “a” na cela Braille, Figura 6.

O botão **vermelho** foi adicionado com a função de percorrer a lista no sentido inverso, apenas para entendimento, acompanhe as tabelas: Tabela 15 e Tabela 16:

⁷ <https://docs.wokwi.com/pt-BR/parts/wokwi-pushbutton>

Tabela 15 – Ações Relacionadas Aos Botões Pressionados

| Botão Azul Pressionado | Execução (JSON) |
|------------------------|-------------------------|
| Primeiro | { "dots": [13,1234,1] } |
| Segundo | { "dots": [13,1234,1] } |
| Terceiro | { "dots": [13,1234,1] } |

Ao pressionar o botão **azul** ou "próximo", no primeiro, segundo e terceiro clique e acionam seus respectivos números de servomotores são acionados para mover ao ângulo configurado

Tabela 16 – Ações Relacionadas Aos Botões Pressionados

| Botão Vermelho Pressionado | Execução (JSON) |
|----------------------------|-----------------------|
| Primeiro | "dots": [13, 1234, 1] |
| Segundo | "dots": [13, 1234, 1] |
| Terceiro | "dots": [13, 1234, 1] |

Ao pressionar o botão **vermelho** ou "voltar", no primeiro, segundo e terceiro clique, os respectivos números de servomotores são ativados e movidos.

Para o botão Branco, tem a função de reiniciar qualquer momento a lista voltando ao início da execução.

Como método de teste e mecanismos que facilite, o DV, que não teria a parte visual para saber quando estiver pronto para a leitura, implementamos o componente *Buzzer*⁸, é fundamental que ele seja passivo, ou seja, que a frequência em Hz seja determinada e garantir que execute tons de diferentes, com atenção ao que diz a [Associação Brasileira de Normas Técnicas \(2015\)](#) em que a compreensão dos sons está associada a diversos fatores que variam desde as limitações físicas, sensoriais e cognitivas do indivíduo até a qualidade do som emitido, incluindo o seu conteúdo, formato, modo de transmissão e a diferença entre o som emitido e o ruído de fundo.

Um som é definido por três propriedades: frequência, intensidade e duração. A capacidade auditiva humana é mais sensível a sons com frequência entre 20 Hz e 20 000 Hz, intensidades entre 20 dB e 120 dB, e duração mínima de 1 segundo. Sons excedendo 120 dB podem causar desconforto, enquanto sons acima de 140 dB têm potencial para provocar sensações dolorosas ([Associação Brasileira de Normas Técnicas, 2015](#)).

O som do *Buzzer* foi utilizado nos seguintes casos:

Código 3 – Código implementação *buzzer*

```

1 int buzzerPin = 26;
2 tone(buzzerPin, 30, 1000);
3 tone(buzzerPin, 200, 1000);
4 noTone(buzzerPin);

```

Fonte: Autores (2023)

⁸ <https://docs.wokwi.com/pt-BR/parts/wokwi-buzzer>

Em relação às configurações e os padrões da [Associação Brasileira de Normas Técnicas \(2015\)](#):

Definimos o pino 26 como a porta no Arduino (`int buzzerPin = 26;`).

No código do simulador wokwi, na representação acima, linha 2, [Código 3](#), implementamos um tom de erro usando a função `tone` (`tone(buzzerPin, 30, 1000);`).

Na linha 3, do [Código 3](#), introduzimos um tom de “ok” utilizando a função `tone` (`tone(buzzerPin, 200, 1000);`).

O parâmetro `buzzerPin` configurado na função `tone()`, e mencionado nas linhas 1 e 2, [Código 3](#), corresponde à designação do pino para a configuração da porta no Arduino. Os valores 30 e 200 denotam as unidades de medida de frequência em Hz (Hertz), enquanto o valor 1000 representa o intervalo de execução, equivalendo, neste caso, a 1000 (ou 1 segundo).

Esses tons têm o propósito de estabelecer um tipo de alarme e identificação voltado para indivíduos com deficiência visual (DV), empregando a audição como meio para enriquecer a orientação durante as etapas.

Por último, a representação da linha 4, [Código 3](#), utilizamos a função `noTone()` com intuito de parar a execução do som. Dentro passamos apenas o Pin, ou neste caso, `buzzerPin` que corresponde a porta 26, estabelecido inicialmente na linha 1, [Código 3](#).

Em que momentos utilizamos os Hz, conferimos na [Tabela 17](#):

Tabela 17 – Frequência (Hz) Das Ações

| Hz | Descrição da Execução |
|-----------|---|
| 30 (Erro) | Falha na conexão |
| 30 (Erro) | Início da lista percorrida |
| 30 (Erro) | Fim da lista percorrida |
| 200 (Ok) | Mensagem recebida através do MQTT |
| 200 (Ok) | Mensagem mapeada e redirecionada para seus servomotores |

Autores 2023

Por outro lado, a linha 2, foi responsável por estabelecer a assinatura. No contexto da API em Python, a biblioteca utilizada foi a `paho` ([Roger Light, 2014](#)), que fornece suporte MQTT. O broker e a porta foram configurados da mesma forma que na conexão de publicação, e o tópico escolhido também foi “/twobrancell”, para permitir a publicação/assinatura do Braille convertido.

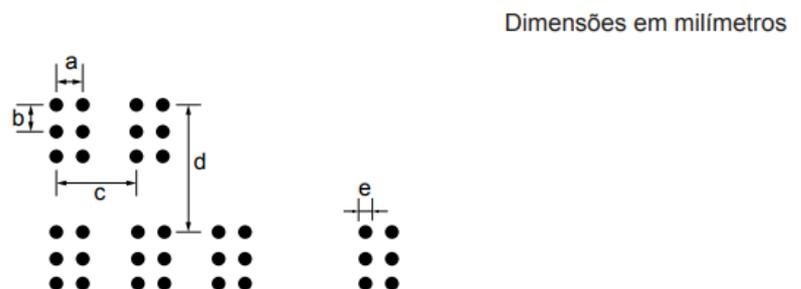
No contexto do Arduino simulado pelo Wokwi, o retorno da mensagem em MQTT foi tratado pelo simulador Arduino, que gerenciou cada caractere do texto recebido, separando-os por conjuntos de pinos que moveriam os servomotores, representando cada ponto em uma célula Braille, como mostrado na [Figura 18](#).

Apesar dos desafios encontrados, a plataforma Wokwi ofereceu uma base sólida para realizar testes e ajustes no projeto, permitindo a visualização e simulação do funcionamento do sistema antes da implementação em ambiente real. Essa abordagem foi fundamental para identificar e resolver problemas, além de aprimorar a interação do usuário e garantir que a solução final fosse eficiente e eficaz na inclusão de deficientes visuais em fóruns de discussão social ([PRADEEPKANDHASAMY; PRIYA; CHELLAPPAN, 2021](#)).

A pesquisa apresentou uma arquitetura promissora com potencial para beneficiar o letramento em Braille de pessoas com deficiência visual. Os estudos relacionados ao contexto do sistema Braille ainda são recentes, e há oportunidades de descobertas adicionais em relação aos estudos desenvolvidos.

A arquitetura utiliza o protocolo MQTT para comunicação entre dispositivos e possibilita a interação por meio de interface visual e tátil, com comandos de voz. O sistema é baseado em nuvem, facilitando a escalabilidade e o acesso remoto. O dispositivo *display* Braille é estruturado com Arduino e componentes, e a fabricação é realizada com uma impressora 3D, como sugestão do projeto [Santiago et al. \(2020\)](#) e [Torres e Ramirez \(2019\)](#) que contém padrões estabelecidos pela [Associação Brasileira de Normas Técnicas \(2015\)](#) como mostra na [Figura 23](#) e [Figura 24](#).

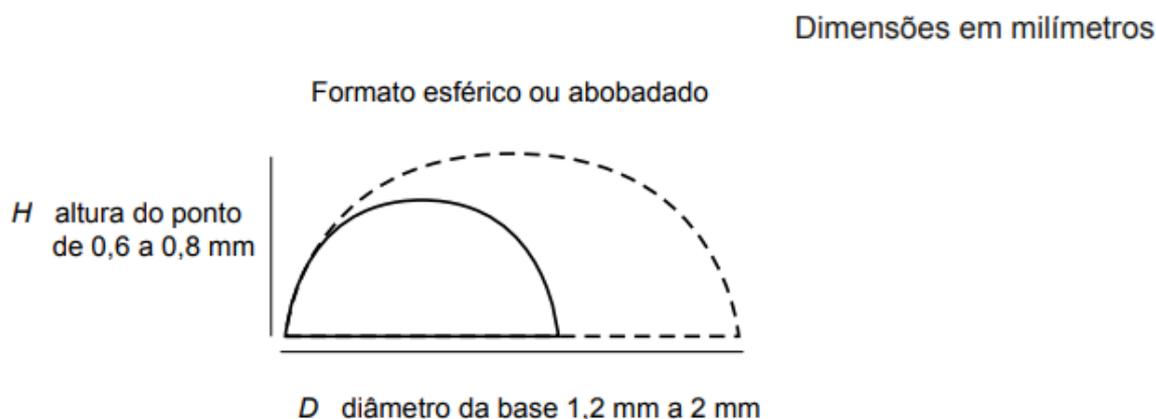
Figura 23 – ABNT: Arranjo geométrico dos pontos em Braille



| a | b | c | d | Diâmetro do ponto e = D | Altura do ponto H |
|-------------------------|-----|-----|------|-------------------------|-------------------|
| 2,7 | 2,7 | 6,6 | 10,8 | de 1,2 a 2,0 | de 0,6 a 0,8 |
| * D significa diâmetro. | | | | | |

Fonte: Autores (2022) cópia editada em (2023)
<https://wokwi.com/projects/348875560981627476>

Figura 24 – ABNT: Formato do relevo do ponto em Braille



Fonte: Autores (2022) cópia editada em (2023)
<<https://wokwi.com/projects/348875560981627476>>

A implementação em nuvem permite o compartilhamento dos dados, com a possibilidade de futuras melhorias. Essa característica o torna ideal para ser utilizado em microcontroladores, especialmente em aplicações de computação de borda que demandam comunicação de dados em alta velocidade entre dispositivos, por meio de uma rede local sem fio *Wireless Local Area Network* (WLAN) (RAMOS-GARCÍA et al., 2022). A arquitetura contribui com a inclusão social e educacional, democratizando o acesso ao conhecimento e tornando a educação mais acessível. No âmbito do protocolo em análise, a questão da segurança não recebeu atenção no presente estudo devido a suas etapas. No entanto, uma abordagem recente apresentada por Shodiq, Pahlevi e Sukarno (2021), foi indicada como uma alternativa viável, mantendo-se fiel à nossa arquitetura que visa aprimorar o acesso ao Braille para deficientes visuais. Qualquer iniciativa de simplificação na utilização é valiosa nesse contexto.

O trabalho de Santiago et al. (2020) propôs um *display* Braille que contextualiza a ideia principal desta arquitetura, embora o uso em nuvem, neste modelo, traga vantagens de escalabilidade e facilidade de acesso em comparação com a instalação em máquinas físicas, favorecendo baixo custo. O seu estudo e o de Torres e Ramirez (2019) trazem propostas que possibilitam o baixo custo na construção de dispositivos táteis como dispositivo de saída para arquitetura, ou também chamado *display* Braille.

Apesar da limitação de tamanho de mensagem do protocolo MQTT, ele ainda pode ser utilizado, desde que a mensagem não exceda esse limite. A implementação em nuvem pode ser adaptada conforme necessário, permitindo começar com planos gratuitos ou básicos e expandir conforme o uso cresce.

Uma das principais vantagens desta arquitetura é permitir que o tutor ou docente use exercícios para diversos alunos, dispensando a necessidade de conhecimentos prévios em Braille, tornando a abordagem mais acessível e inclusiva.

A possibilidade de utilizar interface visual ou tátil e comandos de voz amplia a acessibili-

dade e usabilidade do sistema para os usuários com deficiência visual.

Sugere-se implementar um conversor específico para palavras em português do Brasil em Braille, considerando particularidades como letras e acentos, para aprimorar a experiência dos usuários em contexto brasileiro.

Além disso, é importante realizar um comparativo de desempenho entre outros protocolos, além do MQTT, para solucionar a questão do limite de tamanho das mensagens/JSON e selecionar o protocolo mais adequado para cenários específicos de IoT.

A arquitetura contribui para a inclusão de deficientes visuais em fóruns de discussão social, proporcionando uma experiência de aprendizado acessível, interativa e eficiente em Braille.

A utilização de tecnologias como a impressora 3D, o Arduino e o protocolo MQTT demonstra o potencial da cultura *maker* para criar soluções inovadoras e inclusivas que impactem positivamente a vida das pessoas com deficiência visual.

Este estudo, considera inicialmente a conversão e comunicação via MQTT, com a necessidade de estudos complementares do gerenciamento e controle das publicações e assinaturas.

5.2 Trabalhos futuros

Shodiq, Pahlevi e Sukarno (2021) propõe um mecanismo de autenticação seguro para dispositivos IoT usando o protocolo MQTT. O mecanismo utiliza JSON Web Token (JWT) para autenticação baseada em token e algoritmo de criptografia XXTEA para criptografia de troca de mensagens. O mecanismo é eficaz na prevenção de acesso não autorizado e pode ser implementado em dispositivos restritos.

O XXTEA (*eXtended eXtended Tiny Encryption Algorithm*) é um algoritmo de criptografia simétrica amplamente utilizado em dispositivos com recursos limitados, como IoT, para proteger dados. Especialmente empregado no protocolo MQTT devido à falta de recursos integrados de segurança, ele trabalha em blocos de 32 bits e usa chaves de 128 bits. Por meio de iterações e operações bit a bit, incluindo XOR, adição e deslocamentos, o XXTEA garante a criptografia e descryptografia eficazes dos dados, destacando-se por sua simplicidade, eficiência e resistência a ataques criptográficos conhecidos (SHODIQ; PAHLEVI; SUKARNO, 2021).

A segurança é de suma importância ao lidarmos com a transferência de dados online. No âmbito deste projeto, que também engloba a segurança, o controle dos arquivos gerados pela API, a administração de dispositivos, bem como dos clientes MQTT, propomos uma nova fase neste estudo. O nosso objetivo é simplificar e tornar mais compreensível todo esse processo. É crucial esclarecer, nesta etapa, que caso haja autenticação, estamos plenamente conscientes de que estamos delineando uma arquitetura fundamentada em mecanismos e tecnologias que visam facilitar a usabilidade para dispositivos virtuais.

6

Considerações finais e Trabalhos Futuros

Este capítulo oferece uma visão abrangente dos resultados preliminares da arquitetura desenvolvida para aprimorar o letramento em Braille de pessoas com deficiência visual. A pesquisa aborda um contexto relativamente recente no campo do sistema Braille, com oportunidades significativas para futuras descobertas.

A arquitetura utiliza o protocolo MQTT para comunicação entre dispositivos, permitindo interações por meio de interfaces visuais, táteis e comandos de voz. A base em nuvem possibilita escalabilidade e acesso remoto. O dispositivo Braille é construído com componentes Arduino e fabricado com impressão 3D, seguindo padrões estabelecidos pela norma NBR 9050.

A implementação em nuvem viabiliza o compartilhamento de dados, tornando a arquitetura adequada para microcontroladores e aplicações de computação de borda que exigem comunicação de alta velocidade em redes locais sem fio WLAN. A arquitetura promove a inclusão social e educacional, tornando a educação mais acessível.

A segurança, embora não abordada neste estudo, pode ser aprimorada com abordagens recentes. A arquitetura é acessível e economicamente viável, graças a propostas anteriores.

Apesar do limite de tamanho de mensagem do MQTT, a arquitetura é flexível e pode ser ajustada conforme as necessidades. Ela torna o aprendizado de Braille mais acessível, permitindo que tutores e docentes forneçam exercícios a diversos alunos, independentemente de conhecimento prévio em Braille. A integração de interfaces visuais, táteis e comandos de voz aprimora a acessibilidade.

Recomenda-se a implementação de um conversor específico para o Braille em português do Brasil e a avaliação de outros protocolos para resolver limitações de tamanho de mensagem. A arquitetura promove a inclusão de deficientes visuais em fóruns de discussão social e demonstra o potencial da cultura maker na criação de soluções inclusivas.

Trabalhos futuros incluem a incorporação de mecanismos de segurança para garantir a autenticação segura em dispositivos IoT. A segurança continuará a ser uma prioridade na evolução desta arquitetura.

6.1 Publicação Realizada

• **Architecture Analysis, Cloud-Based Using MQTT Protocol for Braille Literacy.** (QUALIS A3)

SANTOS, J. dos; SILVA, G. J. F. da; RIBEIRO, A. de R. L. Architecture Analysis, Cloud-Based Using MQTT Protocol for Braille Literacy. In: Communication Papers of the 18th Conference on Computer Science and Intelligence Systems, M. Ganzha, L. Maciaszek, M. Paprzycki, D. Ślęzak (eds). ACSIS, Vol. 37, pages 257–263 (2023). Disponível em: <DOI: <http://dx.doi.org/10.15439/2023F8155>>

6.2 Publicação Submetida

• **Promoting Inclusive Learning: Cloud-Based Architecture and MQTT for Braille Literacy.** (QUALIS A1)

SANTOS, J. dos; SILVA, G. J. F. da; RIBEIRO, A. de R. L. Promoting Inclusive Learning: Cloud-Based Architecture and MQTT for Braille Literacy. In: IEEE Internet of Things Journal (2023).

Referências

- Anthony Zhang (Uberi). *SpeechRecognition v.3.8.1*. [S.l.], 2014. Disponível em: <https://github.com/Uberi/speech_recognition/blob/master/reference/library-reference.rst>. Citado na página 57.
- Associação Brasileira de Normas Técnicas. *NBR 9050: Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos*. 2015. Norma Técnica. Citado 3 vezes nas páginas 63, 64 e 65.
- ATAÍDE, J. F.; MESQUITA, N. A. da S. O arborescer das tic na educação: da raiz aos ramos mais recentes. *Revista brasileira de ensino de ciência e tecnologia*, Universidade Tecnológica Federal do Paraná, v. 7, n. 1, 2014. ISSN 1982-873X. Citado na página 30.
- BRASIL, D. F. n. . *Decreto Federal nº 5.296, de 2 de dezembro de 2004*. 2004. Diário Oficial da União. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2004-2006/2004/decreto/d5296.htm>. Citado na página 26.
- BRASIL, L. n. . *Lei nº 10.098, de 19 de dezembro de 2000*. 2000. <http://www.planalto.gov.br/ccivil_03/LEIS/L10098.htm>. Acesso em: 14 de set. de 2021. Disponível em: <http://www.planalto.gov.br/ccivil_03/LEIS/L10098.htm>. Citado na página 26.
- BRASIL, L. n. . *Lei nº 13.146, de 06 de julho de 2015 – Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência)*. 2015. <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm>. Acesso em: 14 de set. de 2021. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm>. Citado 2 vezes nas páginas 22 e 26.
- CALANCEA, C. G. et al. iassistme - adaptable assistant for persons with eye disabilities. *Procedia Computer Science*, v. 159, p. 145–154, 2019. ISSN 1877-0509. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S187705091931347X>>. Citado 5 vezes nas páginas 33, 41, 42, 43 e 56.
- CAPES. *CAPES, P. Portal .periodicos*. 2018. <<https://www.periodicos.capes.gov.br/>>. Disponível em: <<https://www.periodicos.capes.gov.br/>>. Citado na página 39.
- COELHO, C. P.; SOARES, R. G.; ROEHRS, R. Visões sobre inclusão escolar no contexto de educação especial: Pcn x bncc. *Revista Educação e Políticas em Debate*, Maria Vieira Silva, v. 8, n. 2, p. 158–174, 2019. ISSN 2238-8346. Citado 4 vezes nas páginas 16, 18, 27 e 41.
- ECMA International. *ECMAScript® 2017 Language Specification*. [S.l.], 2017. Acesso em: 14 de setembro de 2022. Disponível em: <<https://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>>. Citado 2 vezes nas páginas 31 e 32.
- ESTEVES, P. I. S. et al. SEEstem Wearable navigation device for people with visual impairments. In: Sousa, JP and Henriques, GC and Xavier, JP (Ed.). *ECAADE SIGRADI 2019: ARCHITECTURE IN THE AGE OF THE 4TH INDUSTRIAL REVOLUTION, VOL 1*. DEPT ARCHITECTURE SINT-LUCAS BRUSSELS-GHENT, HOGESCHOOL VOOR WETENSCHAP & KUNST, PALEIZENSTRAAT 65, BRUSSELS, 1030, BELGIUM: ECAADE-EDUCATION & RESEARCH COMPUTER AIDED ARCHITECTURAL DESIGN EUROPE, 2019. p. 681–690. ISBN 978-94-91207-17-4. 37th Conference on

Education-and-Research-in-Computer-Aided-Architectural-Design-in-Europe (eCAADe) / 23rd Conference of the Iberoamerican-Society-Digital-Graphics (SIGraDi), Univ Porto, Fac Architecture, Porto, PORTUGAL, SEP 11-13, 2019. Citado 2 vezes nas páginas 25 e 26.

GANDHI, S.; THAKKER, B.; JHA, S. Braille cell actuator based teaching system for visually impaired students. In: *Braille cell actuator based teaching system for visually impaired students*. Institute of Electrical and Electronics Engineers Inc., 2017. p. 1381–1385. ISBN 9781509007745. Cited By 3; Conference of 1st IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 ; Conference Date: 20 May 2016 Through 21 May 2016; Conference Code:125896. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85015099644&doi=10.1109%2fRTEICT.2016.7808057&partnerID=40&md5=bd3c9373d89ba72173025f1571ee4839>>. Citado 9 vezes nas páginas 27, 28, 41, 42, 43, 44, 45, 56 e 57.

Hubert Pham. *PyAudio 0.2.12*. [S.l.], 2006. Disponível em: <<https://pypi.org/project/PyAudio/>>. Citado na página 57.

IBGE, C. D. . *Censo Demográfico 2010*. 2010. <https://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd_2010_religiao_deficiencia.pdf>. (1) As pessoas incluídas em mais de um tipo de deficiência foram contadas apenas uma vez. (2) Inclusive as pessoas sem declaração dessas deficiências. (3) Inclusive a população sem qualquer tipo de deficiência. Citado 4 vezes nas páginas 12, 16, 22 e 23.

Instituto Brasileiro de Geografia e Estatística (IBGE). Folheto. *Pessoas com deficiência : 2022*. Rio de Janeiro: IBGE, 2023. 15 p. (Coleção Ibgeana). Disponível somente em meio digital. ISBN 9788524045738. Disponível em: <https://biblioteca.ibge.gov.br/visualizacao/livros/liv102013_informativo.pdf>. Citado 2 vezes nas páginas 24 e 25.

JUNIOR, I. M.; VECE, J. P. Contribuições da computação em nuvem como ferramenta pedagógica na educação superior. *The Journal of Engineering and Exact Sciences*, v. 2, n. 3, p. 92–106, 2016. ISSN 2527-1075. Citado na página 16.

KITCHENHAM, B. A.; BUDGEN, D.; BRERETON, O. P. Using mapping studies as the basis for further research - A participant-observer case study. *Inf. Softw. Technol.*, v. 53, n. 6, p. 638–651, 2011. Disponível em: <<https://doi.org/10.1016/j.infsof.2010.12.011>>. Citado na página 37.

LINS, V. C.; MORAIS, A. M. D. Simulação e avaliação de desempenho de uma blockchain para aplicações iot. *Gestão.org*, v. 19, n. 2, p. 169–183, 2021. ISSN 1679-1827. Citado na página 30.

MARTILLANO, D. et al. Tactica: An android-based low-cost assistive tactile device on basic braille notation reading for visually impaired students in sped centers with iot technology. *Journal of Communications*, v. 14, n. 7, p. 593–600, 2019. Cited By 0. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85068749801&doi=10.12720%2fjcm.14.7.593-600&partnerID=40&md5=01bd69a5f2cc4f34803e5dc33f8da965>>. Citado na página 41.

MARTILLANO, D. et al. Pindots: An assistive six-dot braille cell keying device on basic notation writing for visually impaired students with iot technology. In: . [s.n.], 2018. p. 41–47. Cited By 1. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85061288183&doi=10.1145%2f3291078.3291106&partnerID=40&md5=c203e25a2addea05fa4f6ccb554ebe8c>>. Citado 3 vezes nas páginas 18, 42 e 45.

PATRICIO, T. S. et al. Internet das coisas (iot): As consequências da computação ubíqua na sociedade. *Colloquium humanarum*, v. 15, n. 1, p. 83–93, 2018. ISSN 1809-8207. Citado na página 31.

PETERSEN, K. et al. Systematic mapping studies in software engineering. In: *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*. Swindon, GBR: BCS Learning & Development Ltd., 2008. (EASE'08), p. 68–77. Citado na página 37.

Pierre Nicolas Durette and Contributors. *gTTS v. 2.3.0*. [S.l.], 2014. Disponível em: <<http://gtts.readthedocs.org/>>. Citado na página 57.

POSSATO, A. B.; MONTEIRO, P. O. Docentes de tecnologia da informação e comunicação. *Trabalho & educação*, Universidade Federal de Minas Gerais, v. 29, n. 1, 2020. ISSN 1516-9537. Citado 2 vezes nas páginas 30 e 41.

PRADEEPKANDHASAMY, J.; PRIYA, A.; CHELLAPPAN, P. Voice o braille. *Materials Today: Proceedings*, 2021. ISSN 2214-7853. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2214785320407710>>. Citado 10 vezes nas páginas 41, 43, 44, 45, 46, 50, 55, 56, 57 e 64.

Python Core Team. *Python: A dynamic, open source programming language*. [S.l.], 2019. Disponível em: <<https://www.python.org/>>. Citado na página 57.

RAMOS-GARCÍA, O. I. et al. An iot braille display towards assisting visually impaired students in mexico. *Engineering Proceedings*, v. 27, n. 1, 2022. ISSN 2673-4591. Disponível em: <<https://www.mdpi.com/2673-4591/27/1/11>>. Citado 4 vezes nas páginas 19, 31, 41 e 66.

Roger Light. *paho-mqtt 1.6.1*. [S.l.], 2014. Disponível em: <<https://pypi.org/project/paho-mqtt/>>. Citado na página 64.

SALVI, S.; PAHAR, S.; KADALE, Y. Smart glass using iot and machine learning technologies to aid the blind, dumb and deaf. In: . IOP Publishing Ltd, 2021. v. 1804, n. 1. ISSN 17426588. Cited By 1; Conference of 2nd International Conference of Modern Applications on Information and Communication Technology, ICMAICT 2020 ; Conference Date: 22 October 2020 Through 23 October 2020; Conference Code:167555. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85102340654&doi=10.1088%2f1742-6596%2f1804%2f1%2f012181&partnerID=40&md5=d1141b3f452a956112b09317d490e05b>>. Citado 4 vezes nas páginas 42, 43, 44 e 45.

Celta: Sistema de célula tátil para leitura braille (Jogos, Simulações, materiais para editar online). Software. Disponível em: <<http://educapes.capes.gov.br/handle/capes/568113>>. Citado 14 vezes nas páginas 19, 27, 32, 34, 35, 36, 41, 47, 48, 49, 57, 61, 65 e 66.

SHODIQ, F. A.; PAHLEVI, R. R.; SUKARNO, P. Secure mqtt authentication and message exchange methods for iot constrained device. In: *2021 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*. [S.l.: s.n.], 2021. p. 70–74. Citado 3 vezes nas páginas 41, 66 e 67.

SILVA, E. L. d.; MENEZES, E. M. *Metodologia da pesquisa e elaboração de dissertação*. 4. ed. Florianópolis: UFSC, 2005. 138 p. CDU: 001.8. Citado na página 21.

SPOHN, M. A.; BONETTI, F. D. S. Análise de desempenho de uma arquitetura de fog computing para internet of things via estudos de caso. *Revista Brasileira de Computação Aplicada.*, Universidade de Passo Fundo (UPF), v. 11, n. 3, p. 72–87, 2019. ISSN 2176-6649. Citado na página 33.

TORRES, H. D.; RAMIREZ, A. R. G. Projeto de uma cela braille de baixo custo. *Revista Metropolitana de Sistemas e Computação*, v. 3, n. 4, p. 189, 2019. Disponível em: <<https://doi.org/10.37444/issn-2594-5343.v3i4.189>>. Citado 8 vezes nas páginas 19, 34, 35, 41, 47, 61, 65 e 66.

YASSEIN, M. B. et al. Internet of things: Survey and open issues of mqtt protocol. In: *2017 International Conference on Engineering & MIS (ICEMIS)*. [S.l.: s.n.], 2017. p. 1–6. Citado na página 32.

Anexos

ANEXO A – Artigo Publicado

Architecture Analysis Cloud-Based Using MQTT Protocol for Braille Literacy

Josimar dos Santos
Program for Graduate Studies in
Computer Science (PROCC)
Federal University of Sergipe (UFS)
Aracaju-SE, Brazil 49100-000
Email: josimarsts@academico.ufs.br

Gilton José Ferreira da Silva
Program for Graduate Studies in
Computer Science (PROCC)
Federal University of Sergipe (UFS)
Aracaju-SE, Brazil 49100-000
Email: gilton@dcomp.ufs.br

Admilson de Ribamar Lima
Program for Graduate Studies in
Computer Science (PROCC)
Federal University of Sergipe (UFS)
Aracaju-SE, Brazil 49100-000
Email: admilson@ufs.br

Abstract—The use of assistive technologies and social inclusion is becoming increasingly important in both traditional education and active learning methodologies. In innovative education, technologies can play a crucial role in teaching people with disabilities (PWD), offering new perspectives in learning. In this context, this work is specifically aimed at individuals with visual impairments, specifically those with total loss of vision, whether congenital or acquired, or even for individuals interested in learning Braille.

The learning needs in this case are based on sensory perception, such as touch, taste, smell, and hearing. Therefore, the objective of this study is to present a cloud-based architecture that integrates a tactile reading device and Braille display, which is simple, low-cost, and uses a lightweight messaging protocol like the *Message Queuing Telemetry Transport* (MQTT), widely used in Internet of Things (IoT) architectures.

In this architecture, the entire process of converting text into Braille occurs in the cloud. The used device has voice commands issued, received, and processed through an Application Programming Interface (API), commonly known as API, which performs the conversion of text to Braille. Then, the device prepares to display the points in high or low relief in a Braille cell, representing each character of the text. In this way, the visually impaired person can read the character through touch sensitivity.

This architecture could provide a practical and accessible solution for learning Braille and for reading by visually impaired individuals, enabling greater inclusion and active participation in the educational process.

I. INTRODUCTION

THE Information and Communication Technologies (ICT) have become consolidated through the processes of globalization, bringing significant changes and characteristics to the perspective of consumerism that permeate digital and social media. ICTs are present in our daily lives, with many of these technologies already integrated into humanity. We use them often without even realizing the level of evolution we have reached, constantly seeking to create, improve, or correct processes and connections, sending or receiving data, a subject for the next topic. Kenski (2007) states that “technologies are as old as the human species.” Hence, it is essential to reflect on the relationship between education and technology. About the formation of human society with technology [...] [1]. This excessive connectivity allows devices to also communicate,

interact, and work together to achieve a specific objective or make decisions, as in Internet of Things (IoT) sensor networks. Consequently, with the development process, many devices, tools, and computational systems become increasingly capable of being ubiquitous, with dynamic and virtual connections that define the characteristics of Cloud Computing [2].

In this case, how to obtain a basic and low-cost architecture that allows conversion to the Braille system and obtain an interpretable output?

According to the research by [3], one of the highest rates among disabilities – visual, auditory, motor, and mental or intellectual – is visual impairment, with 20.1% in the age range between 15 and 64 years. This rate represents all individuals who declared having some level of difficulty in seeing. However, this study will be inclined to favor people with total loss of vision, but educators and people with sight interested in learning Braille can also benefit.

This study is organized as follows: firstly, aspects related to people with disabilities (PWDs) are discussed, with a specific focus on visual impairment. In addition, topics such as the Internet of Things, MQTT protocol, cloud computing, 3D modeling, maker culture, as well as their implications in the project using Arduino to compose the device, are addressed.

Subsequently, the processes of the research methodology adopted in this study are presented. Then, the proposed architecture is exposed, highlighting its main elements and operation. The results obtained from the application of this architecture are then discussed.

The final considerations of the study are presented, addressing the main insights, conclusions, and possible directions for future work. Finally, the bibliographic references that underpinned and substantiated the study are listed. This structure ensures a comprehensive and systematic approach to the topic, providing a detailed and well-founded analysis of the project at hand.

II. METHODOLOGY

According to the statements by [4], the methodology based on systematic mapping involves planning research with the intention of mapping all literature of empirical and non-empirical studies in a specific thematic area that can be

submitted to accounting, selection, qualification, and finally, data extraction to not only answer the main question but also derived questions. In the following sequence, the authors relate systematic mapping to five process steps in the following order: defining research questions; Conducting Primary Studies Research; Screening articles based on inclusion/exclusion criteria; article classification; data extraction and aggregation.

In this chapter, the processes used to return the results, designated through a systematic mapping conducted between September 2021 and December 2021, are gathered.

A. Internet of Things

For [5], IoT is defined as: “[...] a system of cooperation between connected smart devices.” These devices can be anything, as long as they can connect to the internet, with the purpose of sending and receiving data using the cloud and protocols to compose an “ecosystem.” Defining actions, displaying data, performing tasks. It is with this intention that IoT fits into the architecture of this project. Having a terminal that can receive data, process it, and display it in a way that the user can interpret is the idea we need to “display” the conversion of points into Braille, that is, high and low-relief points so that the user can learn or “read” what is being “displayed.”

B. Cloud Computing

The proposal of cloud computing enables exploring services and obtaining availability, mobility, flexibility, security, sharing, and low infrastructure cost, as reported by [6].

[7] addresses how the infrastructure facilitates the scalability of these resources, and maintenance can be outsourced. “[...] Resources such as processing and storage can be contracted/reserved according to demand.” [7].

C. Modeling, 3D Printing, and Maker Culture

3D printing has proven to be a revolutionary technology that has positively impacted the lives of its users. By offering the possibility of building prototypes and customized objects to meet specific needs, either individually or in collaboration with others, it has opened new creative and practical perspectives [8].

The journey of 3D printing began in 1980 with Chuck Hull, a pioneer in creating stereolithography technology. This process uses laser heating of liquid elements that solidify to form the desired object. The construction with a 3D printer occurs by adding fragments, layer by layer, until the object is completed [8].

Among the various technologies available for building 3D objects, the use of plastic filament stands out, where the filament is melted and injected at specific positions to construct the object in the printer. The general process of creating 3D parts can be summarized in two steps: software modeling and printing of the developed piece.

Currently, there are various accessible software options for 3D modeling, with *AutoDesk® Tinkercad* being an example recommended for beginners due to its usability and free

availability. In this type of modeling, the approach is based on adding and modifying positive and negative three-dimensional shapes, generating or “destroying” content in the physical world [8].

In the context of this project, 3D modeling and printing are fundamental to creating a final product, even if it is a prototype, with a more appealing presentation. The combination of modeling and 3D printing technology allows the development of a *case* structure to protect wires, microcontrollers, and used parts, as well as to build the parts that make up the mechanics of the Braille relief points, for example.

The modernization of the market and accessibility to new technological products, such as the 3D printer, have driven the maker culture. This movement encompasses people who seek to learn concepts from various areas, such as design, carpentry, and technology, through practical experiences. The maker culture encourages individuals to build new products, disassemble objects to understand their manufacturing processes, and thus satisfy their creative and functional needs [8].

An important aspect of the maker movement is the promotion of creative learning through experiences in maker spaces, where people create prototypes and develop new products that meet their specific needs or those of a certain audience [8].

In the context of product prototyping in the maker culture, we can understand that Arduino is a technology widely used to build prototypes for electronic and computational devices. It is important to note that the Arduino technology was developed for rapid prototyping of products for computational physics. In the development of the Arduino technology, we must be clear that it is characterized by three aspects: 1. A software: which is a platform for program development. 2. A legal aspect: which consists of using the technology without worrying about copyright issues. 3. A hardware: boards with electronic components. There are different types of boards when it comes to hardware, just as there are different software options for product development [8].

In summary, 3D printing has played a significant role in promoting creativity and people’s ability to materialize their ideas and needs. The emerging maker culture strengthens this movement, driving creative learning and encouraging the development of personalized and innovative solutions. With technology constantly evolving, we can expect 3D printing to continue to have a positive impact on society, empowering people to turn their ideas into reality.

D. Braille System

In order to understand how the conversion process will be performed and displayed on the Braille display, we need to comprehend the Braille system itself. [9] explain that the formation of a three-by-two (3 rows and 2 columns) matrix of raised dots constitutes a Braille cell, as shown in Figure 1. Different combinations of raised or non-raised dots form representations of letters, numbers, and symbols for tactile mapping. Furthermore, when Braille cells are arranged side by side, they form words [9]. The Braille alphabet can be seen in Figure 2.

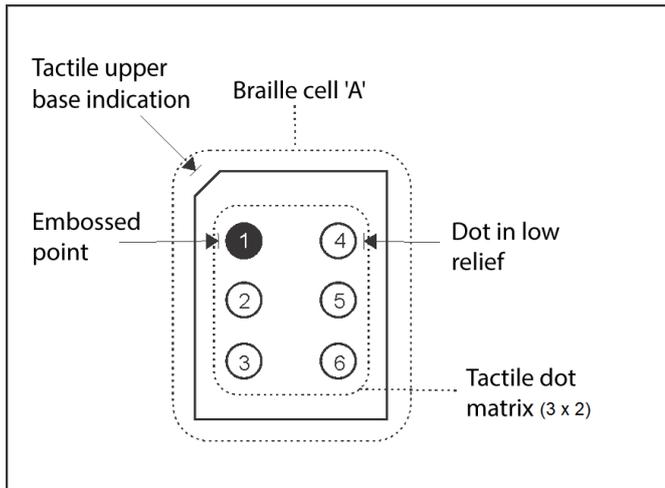


Figure 1. Braille Cell – Representation of the letter A – Authors (2021)

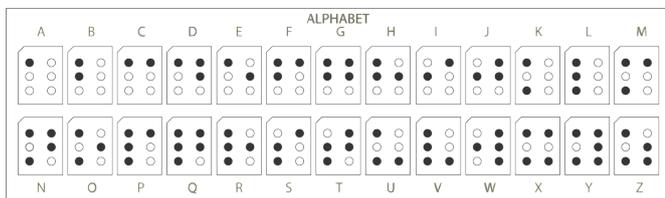


Figure 2. Braille Alphabet – Authors (2021)

E. Structural Concept

The work by [10] aims to transform the inclusion of visually impaired individuals in social discussion forums where transmission is done through speech. With the availability of three main components – Voice to Text, Text to Braille, and Text to Voice - and the ability to save, listen, and delete, according to the presented architecture, the stored file can be converted to Braille and printed.

The study conducted by [9] discusses available technologies that assist visually impaired individuals, ranging from smartphones to tactile Braille input devices and applications. However, the authors note that despite existing innovations, some tutors still use boards with raised or indented points to provide access to the Braille code, requiring more time to teach each student individually and specialized tutor training. [9] propose mass teaching of the Braille code, making the teaching process flexible, fast, and easy to use. The Microcontroller-based Actuator and Cortex-M4 with an Embedded System, the Braille Cell, is a project that allows the tutor to teach multiple students at once.

F. Discussion about the Works

For a better understanding, we have classified subsections that direct the data extracted from each selected study and their respective authors.

1) *Architectures and Applicability:* According to [10], the application is aimed at tutors/presenters who assist visually

impaired individuals in their forums. This system contains the following main functionalities: it recognizes speech and converts it to text during execution, and displays the text on the screen. Once the text is shown on the screen, the user may be able to save the text, listen to the text, review the text, or they can convert this text to Braille. If the text is converted to Braille, the user can print it, as shown in Figures 3 and 4.

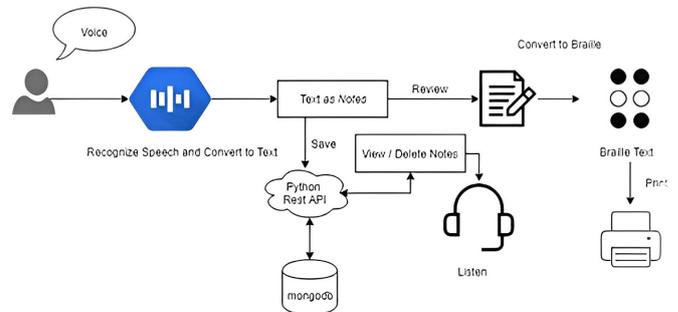


Figure 3. Architecture – Voice o Braille - [10]

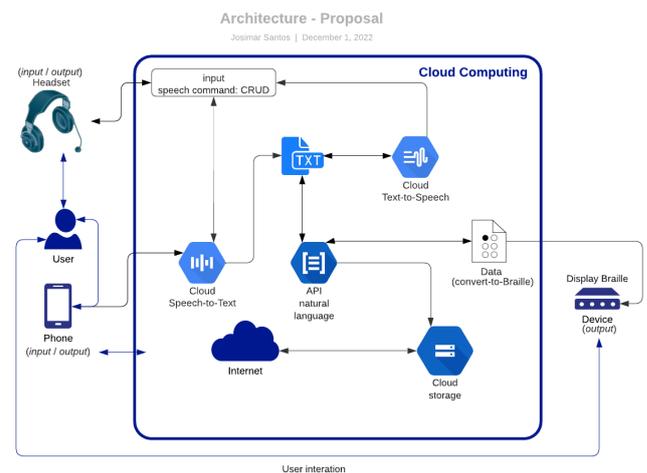


Figure 4. Architecture – Proposal – Authors - 2021

[9] mentions the usage of their project: The teacher will hold the student’s hand and guide their finger on the matrix of raised dots. With the help of instructions and tactile interface, the student gradually learns the Braille code for the alphabet. Therefore, a single tutor can only teach one student at a time, and the tutor needs specialized training to teach Braille codes. This teaching method consumes a considerable amount of time when teaching a group of students, as each student needs to be taught individually. Hence, a concept of mass teaching system is proposed in this article to minimize the time required for teaching. This system has wireless features to avoid the use of cables so that students are not restricted to sitting in any specific position and at a particular distance. The tutor can type the character on the touchscreen keyboard, and this character is wirelessly transmitted to the entire student panel.

In the work of [10], beforehand, their proposal introduces the possibility of storage but not sharing of these Braille notations. Another aspect is that although it allows saving and listening to the saved text, we did not identify the possibility of editing or correcting the stored notations in MongoDB; if there is an error, the only available option is deletion. Another storage possibility, to adapt a modeling based on the studies of [10], could be tested in relation to the studies of [6], which report on low-latency and flexible cloud storage due to serverless architecture.

III. ARCHITECTURE PROPOSAL

In the work of [11], the development of an affordable Braille cell for the local context is discussed. In the initial iterations, the control system kept the solenoids active indefinitely to represent the Braille letters, but this caused excessive energy dissipation in the form of heat, resulting in high energy consumption and risk of damage. To solve this problem, the solenoids were programmed to be activated for only four seconds, allowing the interpretation of symbols and keeping them inactive for the remaining time.

The project faced difficulties in choosing the solenoids, as options with desirable characteristics were scarce in Mexico. In future iterations, hardware components that allow the miniaturization of the Braille cell and reduce energy consumption are analyzed.

In conclusion, the study demonstrated that it is possible to create an affordable Braille cell for the local context. The next steps involve reducing the size of the prototype and conducting tests with users to validate its usability, usefulness, and effectiveness. The work was partially funded by the Human-Computer Interaction Laboratory (IHCLab) of the School of Telematics at the University of Colima, Mexico. There was no need for ethical review for this research.

The study by [12] developed an assistive technology called PINDOTS, with a low-cost and easy-to-use Braille device for students with visual impairment and special education teachers. The technology consists of a six-dot Braille cell and six buttons for basic notation writing. A mobile application was created for teachers, allowing them to send exercises to the Braille device through a wireless connection. The device is capable of spelling three-letter words and has the option to emboss and record Braille dots. To improve the technology, they suggest increasing the number of spelled letters, adding contractions, including numbers, and testing with different types of visual impairments.

The studies by [10] and their architecture are very promising. The architecture of this work provides a vision that guides the structure of the Voice o Braille project, applying some differentials, such as:

Output: Developing a Braille display with the idea of [9], to read characters on Arduino and use it as an output terminal, provides more cost-effectiveness compared to the Braille printer in the author's architecture [10]. As protection for the device, a box, as well as the tactile points, can be 3D printed.

Cloud: Brokers for MQTT publish/subscribe, as well as storage and execution tools for the conversion API, are deployed in the cloud.

The first two works were included after the research and during the experiment, and it is important to cite them as a way to complement the content and add knowledge.

The next steps contextualize and specify how the third-party libraries found and tested in the architecture can assist in each process of the architecture.

The architecture will provide instructions through voice commands, where an "assistant" will speak the process and request commands for decision-making. The user will respond with the desired command to continue the flow. The conversion from voice to text will be done using the SpeechRecognition module [13] in version 3.8.1. The gTTS library [14] in version 2.3.0 will perform the text-to-speech conversion, with the help of the PyAudio library [15] in version 0.2.12. A Python API in version 3.9 will manage a text file with the extension ".txt", enabling resuming the file later to continue, asking the assistant to "read" the text in the file, and also allowing the user to edit parts of the text or delete it. Among these functions, it will also be possible to use the command "convert" to convert the text into a format that will be interpreted by the Braille display device. The API was tested and executed locally using Flask [16], where the results were verified as shown in Figures 5 and 6.



Figure 5. Architecture – API: text conversion screen – authors - 2022

In turn, this Braille display device will be structured with Arduino, specifically an ESP32, servomotors, pushbuttons, among other components for simulation. The display will "show" only one character at a time, and the control to move to the next character can be managed by buttons on the Braille display. For this purpose, we tested platforms like Tinkercad¹, where we did not obtain availability for the ESP32

¹<https://www.tinkercad.com/dashboard>



Figure 6. Architecture – API: converted text – authors - 2022

or simulation functionalities with internet communication due to security reasons.

IV. PRELIMINARY RESULTS

We obtained satisfactory results for initial tests with the Wokwi platform², where we managed to structure the device with the necessary components. However, some delays were noticed while using the libraries for both speech-to-text and text-to-speech conversions.

During the execution of pre-defined voice texts from the “assistant,” if the audio was being generated for the first time, there was a longer delay for the “assistant to speak.” This delay occurs because the strings or phrases mentioned to the user are directly executed in the functions of the libraries used. To overcome this problem, it is possible to store these audios after the first execution, as they are reusable, and play them each time they are called. However, this requires sufficient storage space for these files.

For voice capture from the user, even with the noise reduction applied in the library, some words could not be identified, or the capture was not successful. The words or phrases stored in text files on the server can be converted later and sent in JSON format over the internet to a broker, which can be published by Mosquitto³. The communication was tested through local simulation and the functionalities provided by the Wokwi platform.

One problem encountered regarding the return message in MQTT is that, since the API prepares a JSON with the text of the file requested by the user, we do not know the size of the text that will be converted. As a result, MQTT does not support

sending messages above 268,435,455 bytes, approximately 260 MB. Therefore, a solution to use MQTT would be to limit the number of characters stored in each text file. The JSON is sent and received by the Arduino simulator, which handles the message received via MQTT and manages each character, separating them into sets of pins that will move the servomotors representing each Braille dot, as shown in Figure 7.

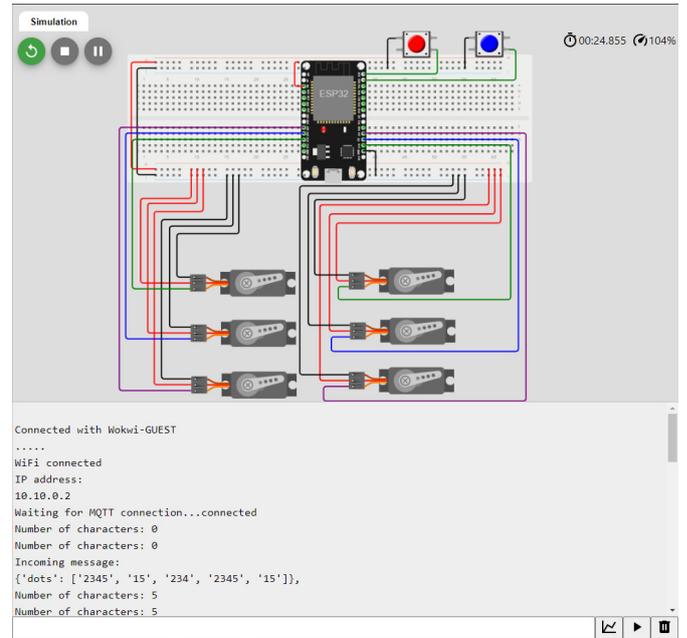


Figure 7. Architecture – Arduino ESP32 Simulator – authors - 2022

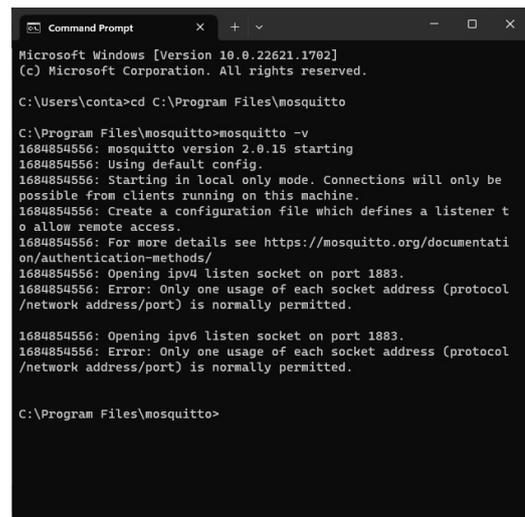


Figure 8. Mosquitto - version 2.0.15 starting

For communication between the server and the Braille dis-

²<https://docs.wokwi.com/pt-BR/>

³<https://mosquitto.org/>

```

Converts text to Braille
Start function localize_notation: 07:36:26
1684838186.656801s
1684838186.656801s
CONVERTER text to Braille - converts the text in the file to Braille
Start function text_to_convert_notation: 07:36:26
1684838186.656801s
1684838186.656801s
[{"hex": ["74", "65", "73", "74", "65"], "ascii": ["t", "e", "s", "t", "e"], "binary": ["001111", "010001", "000111", "001111", "010001"], "dots": ["2345", "15", "234", "2345", "15"], "unicode": ["t", "e", "s", "t", "e"]}]
converted: [{"hex": ["74", "65", "73", "74", "65"], "ascii": ["t", "e", "s", "t", "e"], "binary": ["001111", "010001", "000111", "001111", "010001"], "dots": ["2345", "15", "234", "2345", "15"], "unicode": ["t", "e", "s", "t", "e"]}]
Call endpoint /convert_notation
Start function convert_notation: 07:36:26
1684838186.6588044s
1684838186.6588044s
192.168.0.232 - - [23/May/2023 07:36:26] "POST /convert_notation/ HTTP/1.1" 302 -
192.168.0.232 - - [23/May/2023 07:36:26] "GET / HTTP/1.1" 200 -
192.168.0.232 - - [23/May/2023 07:36:26] "GET /static/images/2BRACELL.png HTTP/1.1" 304 -
192.168.0.232 - - [23/May/2023 07:36:26] "GET /favicon.ico HTTP/1.1" 200 -

```

Figure 9. Architecture – API: send JSON data as a message MQTT to ESP32 simulator – authors - 2022

play, the Wokwi platform, link to the project in development⁴, allowed local simulation and the use of available features, such as the use of the Mosquitto broker⁵ (Figure 8) for MQTT communication. However, a problem was identified regarding the size of the message in MQTT since the API prepares a JSON (JavaScript Object Notation) (Figure 9 and Figure 6), which composes a data structure with the text of the file requested by the user, whose size can exceed the limit supported by MQTT. To solve this issue, we plan to limit the number of characters stored in each text file, ensuring that the messages stay within the supported limit.

The word or phrase is converted to Braille into a structure that will allow future work in various aspects. The database where the letters, numbers, and characters are converted through their respective representations is provided by a *ss-braille.csv* file⁶ available on the Kaggle platform⁷. The file has been carefully updated to meet the Braille conversion needs, such as starting with uppercase letters, as well as words for writing in Brazilian Portuguese, e.g., *c* and *ç*. An example below shows the structure and types of conversion for the word “test”, sent by the API to the display via MQTT. The **hexadecimal** or hex represents the characters in hexadecimal according to

the ASCII table⁸. The **ascii** represents the characters according to the ASCII table⁹. The **binary** sends a set of 6 bits between 0 and 1 for each character of the word, which can be represented by 0 (zero) when the dot is down and 1 when it is raised. The **dots** represent a set of 6 numbers from 1 to 6 that indicate which dot will be raised. For example, if the set includes "12," dots 1 and 2 will be raised, while the numbers "345," absent from the set, represent the dots down.

⁴<https://wokwi.com/projects/348875560981627476>

⁵<https://mosquitto.org/>

⁶<https://www.kaggle.com/datasets/josimarsts/ssbraille?select=ssbraille.csv>

⁷<https://www.kaggle.com/>

⁸<https://www.ascii-code.com/>

⁹<https://www.ascii-code.com/>

The **unicode** represents each character in their respective Braille dots used for display. Regarding display, the article is directed towards an architectural solution that can benefit visually impaired individuals as well as educators and interested parties. So, in addition to enabling communication with the API through voice commands, the architecture provides an interface for conversion. The research project aims to integrate Cloud architecture and the MQTT protocol with IoT Arduino devices, along with the use of Braille for information display. The primary focus is not on commercializing the results but rather on promoting accessibility and inclusion for visually impaired individuals in the context of the Internet of Things. The purpose is to develop a technological solution that allows visually impaired individuals to interact more independently with IoT devices and services, providing them with access to relevant information and improving their quality of life. The research aims to contribute to the advancement of knowledge and offer a practical application that benefits society, especially those with special accessibility needs.

In the scenario where Arduino is simulated through Wokwi, the processing of the MQTT message return is handled by the Arduino simulator itself. It takes care of each character present in the received text and then separates them into groups of pins that, in turn, are responsible for controlling the servomotors. These servomotors represent the dots in a Braille cell, as shown in Figure 7.

Despite the challenges faced, the Wokwi platform provides a solid foundation for testing and adjusting the “Voice o Braille” project, allowing the visualization and simulation of the system’s operation before implementing it in a real environment. This approach is essential to identify and solve problems, as well as to improve user interaction, ensuring that the final solution is efficient and effective in including visually impaired individuals in social and educational forums.

V. CONSIDERATIONS

This work presents an architectural proposal to promote Braille literacy for visually impaired individuals. The architecture uses the MQTT protocol for communication between devices and enables interaction through both visual and tactile interfaces, with voice commands. The system is cloud-based, facilitating scalability and remote access. The Braille display device is structured with Arduino and components, and manufacturing is done with a 3D printer. The cloud implementation allows for data sharing and storage, with the possibility of future improvements, such as a specific converter for Brazilian Portuguese. The proposal aims to promote social and educational inclusion, democratizing access to knowledge and making education more accessible and equitable.

VI. CONCLUSION

The research presented an architecture that can benefit Braille literacy for visually impaired individuals. The related works on the context of the Braille system are recent, and there are still many discoveries that can be made regarding the developed studies. During the research, the study by Santiago

and Bengtson (2020) presented a proposal for a Braille display that contextualizes the main idea of the architecture proposed in this work. However, the authors used a system installed on physical machines and suggested, as future work, the expansion of new possibilities for communication and data sending to the display.

The current project aims to offer MQTT solutions for a Braille display, following the Braille System's standards. This implementation is performed through a specific API developed for this architecture. The API's source code is available on GitHub, allowing other programmers to collaborate and improve the project. In addition, the architecture allows for the creation of other converters according to the specific needs of each project.

The central objective is to provide an MQTT solution for the Braille display that is compatible with Braille System standards, thus ensuring an appropriate experience for users who depend on this reading method. By making the source code available on GitHub, the project becomes open and collaborative, encouraging the participation of other programmers who can contribute with new ideas and improvements.

Another crucial point is the flexibility offered by the architecture, allowing other converters to be implemented according to the specific needs of each project. This makes the solution more versatile and adaptable to different contexts and specific requirements.

In conclusion, the proposed architecture seeks to meet the demands of MQTT communication for the Braille display, following the guidelines of the Braille System and promoting collaboration from the developers' community to continuously improve the project. By implementing new converters according to each case's specific needs, it is hoped that the solution can be widely applicable in different scenarios and contribute to a more inclusive and accessible experience for visually impaired individuals.

Although the MQTT protocol presented limitations regarding the message size, it can still be used as long as it does not exceed this limit. The proposed architecture needs to be analyzed, and depending on its usage, the cloud implementation may vary. However, as cloud computing favors scalability, it is possible to start with free or basic plans and expand as needed. As an additional benefit, the architecture allows tutors/teachers to dispense the need for prior knowledge in Braille and can enable the massive use of exercises for various students, instead of individualized classes, even if the teacher is not familiar with the Braille system. The architecture will enable visual and tactile interaction through the device and voice commands. An implementation of a specific converter in Braille for words in Brazilian Portuguese can be added to the architecture, considering that some letters and accents make the conversion particular in this country. A performance comparison between other protocols, in addition to MQTT, is also considered, following the same architecture suggested in this project to solve the issue of the message size limit/JSON sending. This comparative study would evaluate the performance of these two protocols in terms of message transmission

efficiency and data processing in IoT applications, considering the message/JSON size restriction. This could provide valuable insights for selecting the most suitable protocol in specific scenarios, considering each application's performance and data transmission capacity requirements. The implementation codes will be improved and made available later.

REFERENCES

- [1] A. B. Possato and P. O. Monteiro, "Docentes de tecnologia da informação e comunicação," *Trabalho & educação*, vol. 29, no. 1, 2020.
- [2] I. Machado Junior and J. P. Vece, "Contribuições da computação em nuvem como ferramenta pedagógica na educação superior," *The Journal of Engineering and Exact Sciences*, vol. 2, no. 3, pp. 92–106, 2016.
- [3] C. D. . IBGE, "Censo demográfico 2010," https://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd_2010_religiao_deficiencia.pdf, jul 2010, (1) As pessoas incluídas em mais de um tipo de deficiência foram contadas apenas uma vez. (2) Inclusive as pessoas sem declaração dessas deficiências. (3) Inclusive a população sem qualquer tipo de deficiência.
- [4] B. A. Kitchenham, D. Budgen, and O. P. Brereton, "Using mapping studies as the basis for further research - A participant-observer case study," *Inf. Softw. Technol.*, vol. 53, no. 6, pp. 638–651, 2011. [Online]. Available: <https://doi.org/10.1016/j.infsof.2010.12.011>
- [5] V. C. Lins and A. M. De Moraes, "Simulação e avaliação de desempenho de uma blockchain para aplicações iot," *Gestão.org*, vol. 19, no. 2, pp. 169–183, 2021.
- [6] C. G. Calancea, C.-M. Miluț, L. Alboaie, and A. Iftene, "iassistme - adaptable assistant for persons with eye disabilities," *Procedia Computer Science*, vol. 159, pp. 145–154, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S187705091931347X>
- [7] M. A. Spohn and F. D. S. Bonetti, "Análise de desempenho de uma arquitetura de fog computing para internet of things via estudos de caso," *Revista Brasileira de Computação Aplicada.*, vol. 11, no. 3, pp. 72–87, 2019.
- [8] G. Santiago, C. Bengtson, D. Pino, C. Pendenza, and J. Santos, "Universidade federal de são carlos - ufscar," <https://dcomp.ufscar.br/wp-content/uploads/2016/05/DComp-TR-002.pdf>, 2020.
- [9] S. Gandhi, B. Thakker, and S. Jha, "Braille cell actuator based teaching system for visually impaired students," in *Braille cell actuator based teaching system for visually impaired students*. Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1381–1385, cited By 3; Conference of 1st IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 ; Conference Date: 20 May 2016 Through 21 May 2016; Conference Code:125896. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85015099644&doi=10.1109%2FRTEICT.2016.7808057&partnerID=40&md5=bd3c9373d89ba72173025f1571ee4839>
- [10] J. Pradeepkandhasamy, A. Priya, and P. Chellappan, "Voice o braille," *Materials Today: Proceedings*, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214785320407710>
- [11] O. I. Ramos-García, A. A. Vuelvas-Alvarado, N. A. Osorio-Pérez, M. Ruiz-Torres, F. Estrada-González, L. S. Gaytan-Lugo, S. B. Fajardo-Flores, and P. C. Santana-Mancilla, "An iot braille display towards assisting visually impaired students in mexico," *Engineering Proceedings*, vol. 27, no. 1, 2022. [Online]. Available: <https://www.mdpi.com/2673-4591/27/1/11>
- [12] D. Martillano, A. Chowdhury, J. Delloso, A. Murcia, and R. Mangoma, "Pindots: An assistive six-dot braille cell keying device on basic notation writing for visually impaired students with iot technology," in *Proceedings of the 2018 2nd International Conference on education and e-learning*, ser. ICEEL 2018. ACM, 2018, pp. 41–47.
- [13] Anthony Zhang (Uberi), *SpeechRecognition v.3.8.1*, , 2014. [Online]. Available: https://github.com/Uberi/speech_recognition/blob/master/reference/library-reference.rst
- [14] Pierre Nicolas Durette and Contributors, *gTTS v. 2.3.0*, , 2014. [Online]. Available: <http://gtts.readthedocs.org/>
- [15] Hubert Pham, *PyAudio 0.2.12*, , 2006. [Online]. Available: <https://pypi.org/project/PyAudio/>
- [16] Armin Ronacher, *Flask 2.2.2*, Pallets, 2010. [Online]. Available: <https://pypi.org/project/Flask/>