

Desenvolvimento de um Protótipo de Veículo Automatizado para Automação Logística na Indústria 4.0

Mauricio Pereira Braga, Herysson Rodrigues Figueiredo
Curso de Ciência da Computação
UFN - Universidade Franciscana
Santa Maria - RS
mauricio.pereira@ufn.edu.br, herysson.figueiredo@ufn.edu.br

Resumo—A Indústria 4.0 traz uma transformação nos processos industriais com tecnologias avançadas, como Internet das Coisas e Inteligência Artificial, que impulsionam a automação e a eficiência em operações de manufatura. Este trabalho apresenta uma solução baseada em Veículo Guiado Automatizado para aprimorar o transporte de materiais em ambientes industriais, minimizando a necessidade de intervenção humana e aumentando a segurança. O veículo proposto integra sensores de navegação e sistemas de identificação por *tags*, proporcionando precisão e autonomia na movimentação de cargas. Além do protótipo funcional do veículo, um aplicativo *web* foi desenvolvido para monitoramento e controle. O projeto foi validado em um ambiente industrial simulado, avaliando sua eficácia e aplicabilidade operacional.

Palavras-chave : AGV; Automated Guided Vehicle; Automação Industrial; IoT; RFID; Aplicativo Web; Raspberry Pi; Next.js; MQTT

I. INTRODUÇÃO

A quarta revolução industrial, conhecida como Indústria 4.0, foi inaugurada pelo governo alemão durante a Feira Industrial de Hannover, em 2011 [1]. Esta era é marcada pela integração de tecnologias como a Internet das Coisas (*Internet of Things* - IoT), Sistemas Ciber-físicos e Inteligência Artificial. Estas inovações transformam os ambientes de manufatura, estabelecendo novos padrões para a automação e a eficiência operacional, destacando a interconectividade e a digitalização dos sistemas produtivos [2].

Dentro deste cenário tecnológico, os Veículos Guiados Automatizados (*Automated Guided Vehicles* - AGVs) surgem exemplificando a aplicação prática dos princípios da Indústria 4.0. Utilizados em variados contextos, desde operações de trens sem operador até sistemas complexos de manufatura flexível, os AGVs coordenam o transporte de materiais entre diversos pontos de carga e descarga, integrados a sistemas automatizados para melhorar a logística e o fluxo de trabalho [3]. Contudo, a segurança operacional desses veículos é uma prioridade e, para assegurá-la, é imperativo aderir a normas como a *Verein Deutscher Ingenieure - Automated Guided Vehicles Systems 2510* (VDI 2510) e a Norma Regulamentadora No. 26 (NR-26), que orientam a operação segura dos AGVs, protegendo tanto os trabalhadores quanto as instalações.

Ao implementar uma solução tecnológica que integra sensores para a detecção de obstáculos e linhas para a navegação autônoma dentro de complexos industriais, espera-se mitigar os desafios associados à logística interna. A utilização de Identificação por Radiofrequência (*Radio-Frequency Identification* - RFID) e de sensores de refletância permite não apenas aumentar a precisão de locomoção dentro das instalações, mas também a capacidade de realizar entregas de cargas de maneira autônoma, respondendo dinamicamente a alterações no ambiente de trabalho [4].

A. Justificativa

A fundamentação deste trabalho reside na necessidade de aprimorar a eficiência operacional em ambientes industriais, buscando uma significativa redução da dependência do trabalho manual e uma diminuição dos riscos de erros e acidentes. Nesse contexto, a automação no transporte de cargas emerge como um componente relevante para essa transformação, com potencial não apenas para aumentar a segurança no ambiente de trabalho, mas também para reduzir custos operacionais.

O Veículo Guiado Automatizado proposto foi projetado para enfrentar esses desafios, oferecendo uma solução para o transporte de cargas de um ponto a outro dentro de complexos industriais. Integrando sistemas físicos e digitais, o AGV aprimora a precisão e a agilidade, reduzindo os custos operacionais.

B. Objetivos

1) *Objetivo Geral*: Este trabalho, por meio da automação, tem como objetivo aprimorar a logística em ambientes industriais, desenvolvendo um protótipo funcional de um Veículo Guiado Automatizado, incluindo também o desenvolvimento e a integração de seu aplicativo *web* para gestão e gerenciamento.

2) *Objetivos Específicos*:

- Criar um protótipo de Veículo Guiado Automatizado funcional;
- Realizar navegação por meio de linhas pré-definidas;
- Fazer leitura de *tags* RFID para tomada de decisões;

- Desenvolver um aplicativo *web* para a gestão e gerenciamento do AGV;
- Integrar aplicativo com o protótipo do veículo;
- Testar e validar a eficiência e a segurança do sistema integrado em um ambiente industrial simulado.

Este trabalho é estruturado em várias seções essenciais para a compreensão e desenvolvimento do projeto. A seção subsequente, o Referencial Teórico, explora os conceitos-chave e as inovações tecnológicas da Indústria 4.0. Em seguida, a revisão de Trabalhos Correlatos apresenta estudos sobre AGVs e tecnologias relacionadas, fundamentais para este projeto. A Metodologia utiliza o método Kanban para a gestão do desenvolvimento. As seções de Resultados e Conclusão discutem o planejamento detalhado e fornecem uma visão completa do projeto.

II. REFERENCIAL TEÓRICO

Nesta seção, apresentam-se os conceitos-chave e as inovações tecnológicas que fundamentam o desenvolvimento deste trabalho, enfatizando o papel dos Veículos Guiados Automatizados e suas aplicações para melhorar a eficiência logística e operacional, juntamente com as normas de segurança que regem sua implementação. Também foram pesquisadas as tecnologias de automação, controle e programação e como elas estão interconectadas para criar sistemas mais inteligentes e autônomos.

A. Automação Industrial

A automação industrial utiliza tecnologias de controle e informação para aumentar a eficiência, precisão e segurança nos processos, substituindo práticas manuais por sistemas automatizados e impactando produtividade e economia [5]. Desde os anos 1970, com a introdução dos microprocessadores, o setor evoluiu, incorporando Robótica e sistemas de manufatura flexíveis, além de IoT e Inteligência Artificial [6].

Na Indústria 4.0, a automação integra Sistemas Ciberfísicos¹, IoT², computação em nuvem e *big data*. Essa convergência tecnológica permite fábricas inteligentes, onde sistemas monitoram e tomam decisões de forma autônoma [9].

Nesse cenário, Veículos Guiados Automatizados são destacados como exemplos da adoção de controle avançado e Robótica, desempenhando um papel fundamental na modernização das operações industriais.

B. Automated Guided Vehicles (AGVs)

Os AGVs, ou Veículos Guiados Automatizados, são tecnologias de transporte de materiais que utilizam veículos

autônomos para seguir rotas predefinidas em locais como armazéns e fábricas. Sua principal vantagem é a operação sem trilhos físicos, permitindo uma integração menos intrusiva nos espaços de trabalho [3]. A história dos AGVs começou em 1953 nos Estados Unidos, com o desenvolvimento inicial caracterizado por sistemas de orientação simples e sensores mecânicos. Em 1954, a Barrett-Cravens, em Illinois, automatizou veículos de carga, culminando no uso dos primeiros AGVs pela Mercury Motor Freight na Carolina do Sul, o que marcou o início da automação no transporte de cargas [10].

Atualmente, os AGVs são classificados em três tipos principais [3]:

- Veículos de Reboque: Usados para transportar cargas pesadas ou volumosas por longas distâncias sem operador.
- Caminhões de Paletes: Projetados para movimentar cargas paletizadas em rotas específicas dentro de um espaço.
- Transportadores de Carga Unitária: Concebidos para transportar uma ou mais cargas unitárias por longas distâncias; este modelo foi o escolhido para o projeto.

Com o avanço dos AGVs, a implementação de normas e regulamentações torna-se essencial para garantir a segurança e eficiência dos trabalhadores que compartilham espaços com esses veículos, visando a prevenção de acidentes e assegurando condições operacionais seguras e eficientes.

C. Segurança e Eficiência Operacional

A segurança e a eficiência operacional são essenciais ao introduzir tecnologias avançadas em ambientes industriais, especialmente na adoção de AGVs. Normas como a VDI 2510 e NR-26 são fundamentais para orientar essa adoção, assegurando a segurança dos colaboradores e a proteção das instalações.

1) *Verein Deutscher Ingenieure - Automated Guided Vehicles Systems 2510 (VDI 2510)*: A VDI 2510 é uma norma técnica alemã, que especifica requisitos para a operação segura de AGVs. Entre esses, destaca-se a limitação da velocidade máxima a 1 metro por segundo (m/s), ou 3,6 quilômetros por hora (km/h), para proteger tanto as pessoas quanto as estruturas nas áreas de operação dos AGVs [11].

2) *Norma Regulamentadora No. 26 - Sinalização de Segurança (NR-26)*: A NR-26 recomenda o uso da cor amarela para equipamentos de transporte em geral, visando à segurança. Embora não especifique os AGVs diretamente, essa orientação é aplicável a todos os veículos industriais, incluindo AGVs, para melhorar sua visibilidade e, conseqüentemente, a segurança operacional [12].

A integração de tecnologias de controle e automação validadas é crucial para atender às normas mencionadas. Tal abordagem não somente protege a segurança dos trabalhadores e as instalações, mas também promove a eficiência operacional. Assim, o emprego de AGVs em ambientes

¹Sistemas que combinam processos computacionais e físicos, usando sensores e atuadores para interagir com o ambiente [7].

²A Internet das Coisas conecta dispositivos físicos à internet, facilitando coleta e compartilhamento de dados [8].

industriais demanda um comprometimento com segurança e eficácia, recorrendo a soluções tecnológicas de ponta.

D. Tecnologias para Automação e Controle

O segmento de tecnologias para automação e controle abrange diversas ferramentas que otimizam processos através de comunicação à distância, identificação rápida e processamento de dados em tempo de execução. Destacam-se tecnologias como, Raspberry Pi, RFID, além de sensores de refletância e ultrassônico.

1) *Raspberry Pi*: Raspberry Pi é um computador de placa única de baixo custo, o Raspberry Pi ganhou ampla adoção em projetos de automação e controle devido à sua robusta capacidade de processamento e versatilidade para a integração de diversos sensores e módulos. Destaca-se em diversas áreas, incluindo a educação em ciência da computação, IoT, Robótica e automação, onde desempenha um papel crucial na construção de sistemas inteligentes e conectados [13].

2) *Radio-Frequency Identification (RFID)*: A Identificação por Radiofrequência, em português, permite a comunicação e transmissão de dados por meio de sinais de radiofrequência, facilitando o monitoramento, a contagem e a indexação de objetos sem necessidade de contato físico ou linha de visão direta. O sistema RFID é composto essencialmente por duas partes: as etiquetas RFID, que são afixadas aos objetos a serem identificados, e os leitores RFID, que capturam e interpretam os sinais emitidos pelas etiquetas [14].

3) *Sensor de Refletância*: O Sensor de Refletância é empregado primordialmente para a detecção de linhas e objetos próximos, opera por meio da emissão de luz infravermelha e da medição da quantidade de luz que é refletida de volta por diferentes superfícies. Graças a um fototransistor, diferencia superfícies claras das escuras com base na intensidade da luz refletida [15], característica essencial para a navegação precisa e eficiente de robôs ao longo de trajetórias pré-definidas.

4) *Sensor Ultrassônico*: O Sensor Ultrassônico utiliza o princípio do sonar para calcular a distância até objetos, de maneira similar à empregada por morcegos e golfinhos, o sensor ultrassônico oferece medições de distância precisas e estáveis sem necessidade de contato físico. Resistente a interferências de luz solar e materiais de cores escuras. Seu *design* inclui módulos transmissor e receptor ultrassônicos [16].

Para o pleno funcionamento das tecnologias de automação e controle, é essencial integrá-las adequadamente com tecnologias de programação. Essa integração é crucial para alcançar a máxima eficiência e precisão.

E. Tecnologias para Programação

Na programação de sistemas automatizados, diversas tecnologias desempenham papéis cruciais. Python é valorizado

por sua robustez e versatilidade. *Hypertext Transfer Protocol (HTTP)* e *Message Queuing Telemetry Transport (MQTT)* são importantes para a comunicação entre sistemas, especialmente na *web* e em aplicações de IoT. No desenvolvimento *web*, tecnologias como React, Next.js e o *TypeScript Remote Procedure Call (tRPC)* são relevantes para construir interfaces dinâmicas e facilitar a comunicação com o servidor.

1) *Python e MicroPython*: Python é uma linguagem de programação de alto nível, interpretada e de propósito geral, destacada por sua facilidade de leitura e escrita. Ela suporta múltiplos paradigmas de programação, incluindo programação orientada a objetos, imperativa e funcional, o que a torna versátil para uma ampla gama de projetos, desde aplicações *web* até ciência de dados e inteligência artificial. Criada com o objetivo de ser poderosa e acessível [17], Python enfatiza a legibilidade do código, optando por uma sintaxe clara em vez de complexidade. MicroPython é uma implementação ágil e eficiente da linguagem de programação Python na sua versão 3, otimizada para microcontroladores e ambientes com recursos limitados, trazendo um subconjunto da biblioteca padrão do Python. Sua concepção visa facilitar o desenvolvimento em sistemas embarcados, ao mesmo tempo em que se esforça para manter a máxima compatibilidade com o Python padrão. Isso permite transferir código com facilidade do ambiente de desenvolvimento *desktop* para microcontroladores ou sistemas embarcados [18].

2) *Hypertext Transfer Protocol (HTTP) e Message Queuing Telemetry Transport (MQTT)*: O HTTP é um protocolo de aplicação amplamente utilizado para a transmissão de documentos hipermídia, na *web*. Ele segue o modelo cliente-servidor, onde um cliente faz uma requisição e espera a resposta do servidor. Sendo um protocolo sem estado, HTTP não mantém informações entre requisições, o que requer mecanismos adicionais para sessões persistentes. Embora frequentemente baseado em *Transmission Control Protocol/Internet Protocol (TCP/IP)*, pode operar em qualquer camada de transporte confiável [19]. Por outro lado, MQTT é o protocolo preferido para a IoT, definido por regras que permitem a publicação e assinatura de dados por dispositivos IoT. Orientado a eventos, MQTT facilita a comunicação entre dispositivos dissociados através de tópicos, mediada por um *broker* MQTT que distribui as mensagens corretamente. Sua arquitetura eficiente e de baixo consumo de recursos é ideal para ambientes IoT, garantindo comunicação em tempo de execução com mínima carga de rede [20].

3) *React e Next.js*: React é uma biblioteca JavaScript desenvolvida pela Meta para construir interfaces de usuário de forma eficiente, usando componentes reutilizáveis e

um *Virtual Document Object Model*³ (Virtual DOM) para otimizar as atualizações da interface [22]. Next.js, criado pela Vercel, é um *framework* que amplia o React, oferecendo renderização no servidor, geração de sites estáticos e melhor desempenho inicial das páginas [23].

4) *TypeScript Remote Procedure Call (tRPC)*: O tRPC é uma biblioteca para a linguagem TypeScript que permite a criação de *Application Programming Interface*⁴ (APIs) totalmente tipadas, facilitando a comunicação entre cliente e servidor sem a necessidade de código *boilerplate* ou geração de código adicional. Ao integrar-se com *frameworks* como Next.js, o tRPC permite que os desenvolvedores definam rotas de API e tipos de dados em um único lugar, compartilhando automaticamente esses tipos entre as camadas de cliente e servidor [25].

A fim de garantir a eficiência e precisão no desenvolvimento de sistemas automatizados, é essencial integrar as tecnologias com práticas emergentes e comprovadas. A seguir, são apresentados trabalhos correlatos que demonstram a aplicação prática de várias tecnologias, especialmente no desenvolvimento de Veículos Guiados Automatizados, contribuindo significativamente para os avanços deste trabalho.

III. TRABALHOS CORRELATOS

Nesta seção, são apresentados os artigos relacionados ao uso e desenvolvimento de AGVs e suas tecnologias associadas, que contribuíram para o desenvolvimento deste projeto. Os artigos apresentados nesta seção trazem informações sobre o uso de RFID para navegação e reconhecimento, ideias de modelagem da estrutura de um protótipo de AGV, e, por fim, a conexão entre AGVs e IoT.

A. Veículos Guiados Automatizados Inteligentes para Manufatura no Contexto da Indústria 4.0

Jasprabhjit Mehami, Mauludin Nawi e Ray Y Zhong [26], exploram a adoção de AGVs em fábricas inteligentes. Equiparam dois AGVs com leitores RFID para navegação e reconhecimento de etiquetas no solo e em objetos, aprimorando a logística interna.

Os resultados alcançados demonstraram ser notáveis, comprovando que a integração de Veículos Guiados Automatizados com a tecnologia de RFID pode aprimorar consideravelmente a eficiência operacional em ambientes fabris.

Destaca-se que a integração entre RFID e AGVs representa a principal contribuição do artigo para o desenvolvimento do protótipo deste trabalho.

³O Virtual DOM é uma representação idealizada da interface de usuário, armazenada na memória e sincronizada com o DOM real por meio de bibliotecas como ReactDOM. Esse processo, conhecido como reconciliação, permite uma abordagem declarativa no React, definindo estados desejados para a interface enquanto o React alinha o DOM a esses estados [21].

⁴Uma API ou interface de programação de aplicativo é um conjunto de regras ou protocolos que permite que aplicações de *software* se comuniquem entre si para trocar dados, funcionalidades e recursos [24].

B. Modelagem e Implementação de Protótipo de um Veículo Guiado Automatizado para Fábricas Inteligentes

Hiroki Sasamoto *et al.* [27], investigam a modelagem e implementação de um AGV para transporte de materiais. O *design* utiliza tração diferencial e chapas de aço cortadas a laser, garantindo robustez estrutural capaz de suportar até 700 kg, confirmada por Análise de Elementos Finitos.

Os testes realizados validaram a eficácia do AGV em navegar automaticamente pela fábrica, transportando cargas pesadas e evitando obstáculos. No entanto, o principal elemento que agregou ao desenvolvimento deste trabalho foi a robustez da estrutura mecânica do protótipo.

C. Sistema AGV baseado em IoT para Assistência Médica

Ananthi K *et al.* [28] apresentam um AGV adaptado para ambientes hospitalares, visando reduzir o contato físico durante a pandemia de COVID-19. O AGV realiza distribuição de medicamentos, medições vitais e permite consultas via videochamada, transmitindo informações aos profissionais de saúde via IoT.

O artigo finaliza afirmando que o sistema AGV vai além de apenas atender à necessidade urgente de minimizar os riscos de contágio durante a pandemia. Ele se integra de forma eficaz às operações diárias dos hospitais. Essa capacidade de integração operacional se destacou como a característica mais influente para o presente trabalho, aprimorando a sinergia entre máquinas e sistemas.

Os trabalhos correlatos apresentados fornecem uma base sólida para este trabalho, orientando a escolha de metodologias e práticas de trabalho. As metodologias ágeis, como o Kanban, que incentivam a prototipação, são particularmente relevantes, pois permitem adaptações rápidas e incrementais, promovendo a eficiência e a melhoria contínua dos processos.

IV. METODOLOGIA

A metodologia ágil surgiu para superar as limitações dos métodos tradicionais de desenvolvimento de *software*, que eram burocráticos e pouco flexíveis, principalmente para projetos menores. Em vez de um planejamento rígido e documentação extensa, a metodologia ágil promove um desenvolvimento incremental, adaptando-se a mudanças nos requisitos [29]. Para este trabalho, foi adotado o método Kanban.

O método Kanban é uma abordagem de gestão inicialmente desenvolvida na Toyota e posteriormente adaptada ao campo do desenvolvimento de *software* por David Anderson. Essa metodologia foca na melhoria constante dos processos e fluxos de trabalho, com ênfase na gestão de mudanças e na entrega eficaz de serviços. Uma das principais ferramentas do Kanban é o quadro Kanban, que visualiza o fluxo de trabalho e facilita a gestão das tarefas através de diferentes estágios do processo de desenvolvimento [30].

Tabela II
REQUISITOS NÃO FUNCIONAIS

RNF01	O sistema <i>web</i> deve facilitar a realização de diagnósticos e reparos remotamente.
RNF02	O AGV deve cumprir com todas as normas de segurança aplicáveis (VDI 2510 e NR-26).
RNF03	O <i>design</i> mecânico do AGV deve minimizar riscos de acidente, incluindo proteções adequadas contra uso indevido intencional.
RNF04	O sistema deve registrar todas as ações do AGV.

C. Projeto

Na fase inicial do projeto, foi criado um diagrama de domínio para esclarecer as funcionalidades e relações do sistema, destacando as principais classes: AGV, com atributos como status e métodos de operação; Usuário, que supervisiona o AGV; Rota e Transporte, que definem percursos e listam rotas; Carga, para especificar os itens transportados; e Registro de Atividades, que mantém um *log* de eventos para análise. Esses elementos formam a base do sistema, representados de maneira visual na Figura 2, que ajuda a compreender suas interdependências.

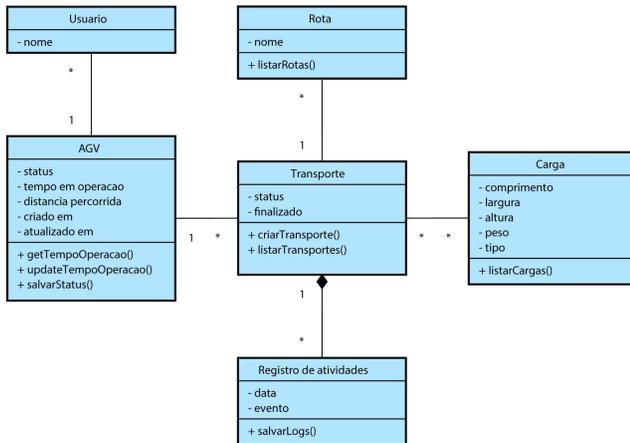


Figura 2. Diagrama de domínio [31].

Para detalhar a operacionalidade do AGV, utilizou-se um diagrama de máquina de estados. Esse diagrama descreve os principais estados do veículo, como "Desligado", "Ligado", "Operação", "Manutenção" e "Travado". Durante a "Operação", o AGV alterna entre os subestados "Transportando" e "Retornando", dependendo da presença de carga. A representação também destaca transições desencadeadas por eventos, como "desligar()", "erro()" e "rotaFinalizada()", conforme ilustrado na Figura 3. Essas transições incluem ações específicas, como acender as luzes verde no estado "Ligado" para sinalizar que o AGV está ligado.

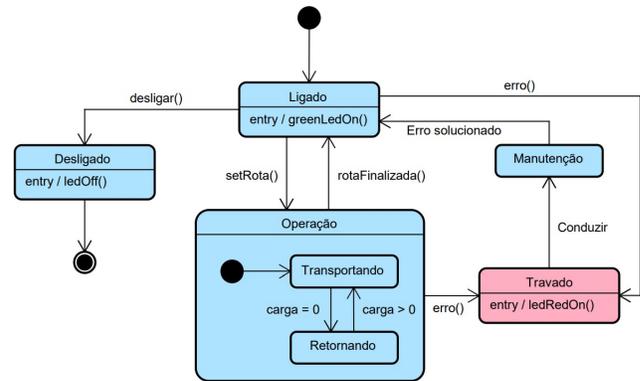


Figura 3. Diagrama de máquina de estado [31].

Além de conceitos abstratos, o projeto materializou-se no modelo tridimensional exibido na Figura 4. Esse modelo foi concebido para transportar cargas unitárias de até 100 mm de comprimento e 70 mm de largura, com altura para suporte de 15 mm. Por outro lado, o AGV em si possui dimensões de 200 mm de comprimento, 150 mm de largura e 52 mm de altura, atendendo às exigências de eficiência em ambientes industriais.

O cuidado com normas e segurança também foi fundamental no *design* mecânico. O projeto segue a NR-26, que requer a cor amarela para garantir visibilidade. Essa escolha contribui para a segurança dos operadores, minimizando o risco de acidentes no ambiente de trabalho.

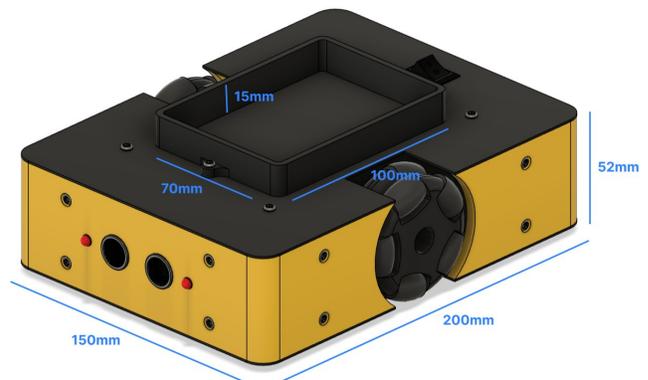


Figura 4. Desenho tridimensional do AGV [31].

Com o projeto concebido, o próximo passo foi dar forma a essas ideias por meio da construção do protótipo. Essa etapa abrangeu a fabricação do modelo físico, ajustes no design inicial, a integração de componentes eletrônicos e o desenvolvimento e integração com o aplicativo *web*, garantindo a comunicação e o controle do sistema.

D. Construção

A construção do protótipo foi realizada por meio de modelagem por fusão e deposição (FDM⁵). Esse processo permitiu ajustes no *design*, como o aumento da largura de 150 mm para 180 mm, a fim de acomodar os componentes internos. Além disso, a altura total foi reduzida para 47 mm, e tampas mais espessas foram produzidas para melhorar a resistência. A estrutura mecânica do AGV é feita predominantemente de plástico, escolhido por sua leveza e adequação ao projeto, conforme ilustrado na Figura 5.

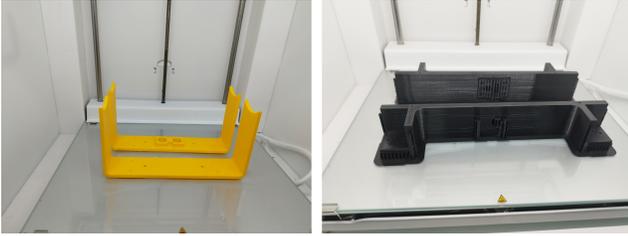


Figura 5. Processo de fabricação das peças [31].

Com a estrutura física concluída, foi implementado o sistema eletrônico, que utiliza uma placa Raspberry Pi programada em MicroPython. A placa toda lógica do veículo, um sensor ultrassônico para detectar obstáculos e um leitor RFID para identificar pontos de referência no ambiente. A Figura 6 apresenta o esquema eletrônico com os componentes principais: o leitor RFID (A), o Raspberry Pi (B), a bateria (C), o sensor ultrassônico (D), os motores (E) e o *driver* de motores (F).

Embora o *driver* seja representado no diagrama como uma ponte H LN298, a aplicação real utiliza um Controlador de Velocidade Eletrônico com regulador de tensão integrado. Essa configuração atende à necessidade de alimentar os motores com 12V e o Raspberry Pi com 5V.

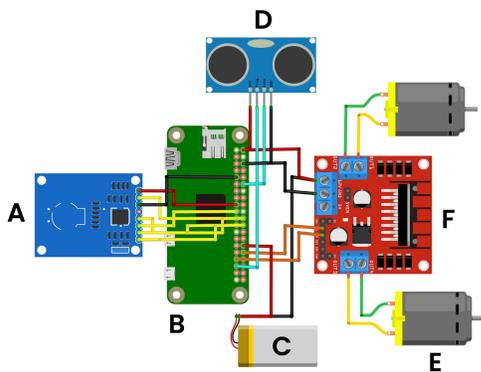


Figura 6. Esquema Eletrônico [31].

⁵FDM, sigla em inglês para *Fused Deposition Modeling*, é um processo de impressão 3D que cria objetos tridimensionais pela deposição camada por camada de material termoplástico fundido [32].

O funcionamento do AGV é gerenciado por um algoritmo que define a rota a ser seguida com base no destino especificado. Como apresentado no *Listing 1*, o método recebe a sigla do destino como entrada e ajusta a sequência de deslocamento do AGV. A lógica utiliza pontos pré-determinados, onde "CRG" representa a área de carga, "ITM" é a interseção do meio, "ITA" e "ITB" são as interseções A e B, e "DGA" ou "DGB" são as áreas de descarga A e B. O símbolo "#" indica o final do percurso.

A função define uma rota específica de acordo com o destino informado pelo usuário, ajustando a sequência de pontos a serem percorridos pelo AGV. O símbolo "#" também é utilizado para indicar o final da rota e possibilita o retorno ao ponto inicial ao final do percurso.

```
1 def definir_rota(self, destino):
2     if destino == "DGA":
3         self.rota = ["CRG", "ITM", "ITA", "DGA", "#"]
4     elif destino == "DGB":
5         self.rota = ["CRG", "ITM", "ITB", "DGB", "#"]
6     else:
7         raise ValueError("Destino invalido!")
8     self.destino = destino
```

Listing 1. Método para definição de rota.

Por fim, o sistema foi integrado a um aplicativo *web* desenvolvido em Next.js com a biblioteca tRPC, formando uma aplicação do tipo *end-to-end*. Essa arquitetura unifica tanto o lado do cliente quanto o do servidor, permitindo uma integração direta entre as camadas. O servidor conecta-se a um banco de dados relacional, assegurando a persistência e a recuperação dos dados relacionados ao AGV.

A comunicação entre os componentes do sistema ocorre de duas maneiras principais. A troca de dados entre cliente e servidor é realizada via HTTP, enquanto a comunicação em tempo de execução entre o Raspberry Pi e o cliente utiliza o protocolo MQTT. O *Listing 2* apresenta o método responsável por publicar o destino do AGV no tópico do MQTT. Esse método traduz o destino especificado em uma mensagem codificada e a publica no tópico "transporte/iniciar", permitindo que o Raspberry Pi receba e processe a informação.

```
1 function publicarDestino(destino: string) {
2     const mensagem =
3         destino === "Descarga A" ? "DGA":
4         destino === "Descarga B" ? "DGB" : "";
5
6     if (mensagem) {
7         mqttClient.publish("transporte/iniciar", mensagem,
8             (error) => {
9             if (error) {
10                console.error("Erro ao publicar destino:", error);
11            } else {
12                console.log(`Destino publicado: ${mensagem}`);
13            }
14        });
15    }
16 }
```

Listing 2. Método para publicar o destino via MQTT.

O processo de construção serviu para integrar de forma organizada os componentes mecânicos, eletrônicos e de

software, assegurando que todos funcionassem de maneira coerente para alcançar os resultados esperados.

V. RESULTADOS

Inicialmente, propôs-se a construção de um protótipo de Veículo Guiado Automatizado capaz de seguir linhas predefinidas, tomar decisões a partir de etiquetas RFID e integrar-se a um aplicativo *web* para validar o conjunto de integrações.

Durante o desenvolvimento, o sensor de refletância apresentou diversas limitações funcionais, levando ao seu desuso no projeto. As imprecisões nas medições realizadas pelo sensor, somadas aos problemas de compatibilidade com o controlador eletrônico de velocidade, comprometeram a eficácia do sistema e inviabilizaram sua utilização. Para aprimorar os resultados, optou-se por usar exclusivamente o leitor RFID para guiar o veículo no trajeto.

Essa mudança trouxe limitações, como um limite fixo no peso da carga. Isso se deve à forma como o veículo realiza as curvas para o destino, controladas por tempo; cargas mais pesadas influenciam o desempenho dos motores e, conseqüentemente, a posição em que o AGV para após uma curva. Apesar dessas limitações, a abordagem com o leitor RFID permitiu uma operação mais consistente.

O AGV foi construído em sua versão final baseada no desenho da Figura 4, com pequenas alterações realizadas durante o processo de fabricação para aprimorar a estabilidade e organização interna. O protótipo conta com seis rodas: duas principais, acopladas aos motores, para tração, e quatro rodízios de apoio posicionados nas extremidades inferiores do veículo. O leitor RFID foi instalado na parte inferior frontal. Na Figura 7, é possível observar o veículo com as luzes verdes acesas, indicando que está ligado.



Figura 7. Versão final do protótipo do AGV [31].

Nos testes de transporte de cargas, considerando uma carga de 92 mm de comprimento, 68 mm de largura, 42 mm de altura e peso aproximado de 10 g, o AGV completou o percurso em cerca de 6 segundos. A Figura 8 ilustra o trajeto realizado do ponto de carga até a área de descarga A, passando pela interseção central e pela interseção A. Em

cargas acima de 800 g, notou-se dificuldade nos motores para realizar curvas e alinhar-se após elas, evidenciando a necessidade de melhorias para cargas mais pesadas.

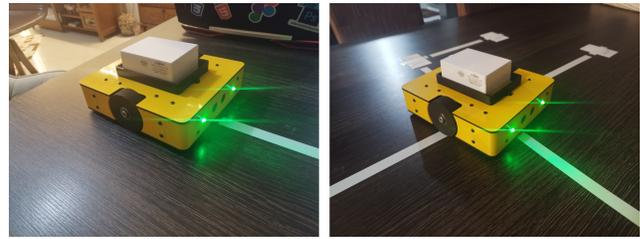


Figura 8. AGV realizando o percurso [31].

Para aumentar a segurança operacional, o sistema anti-colisão foi validado. O sensor ultrassônico, programado para parar o veículo ao detectar um objeto a 10 cm, funcionou de forma eficaz em trajetos retos. Na Figura 9, é possível observar dois objetos usados nos testes e as luzes vermelhas do AGV acesas, indicando alerta. No entanto, o sensor demonstrou limitações ao detectar objetos fora do ângulo frontal, devido à curta distância programada e ao modelo simples utilizado.



Figura 9. Testes de segurança com o sensor ultrassônico [31].

O monitoramento do AGV foi realizado por meio de um aplicativo *web* desenvolvido em Next.js e integrado ao Raspberry Pi via protocolo MQTT, garantindo comunicação em tempo de execução. A Figura 10 ilustra os principais componentes da tela de Visão Geral do aplicativo, como o número de transportes realizados, velocidade, status, tempo, alertas, destino e a localização atual no percurso.

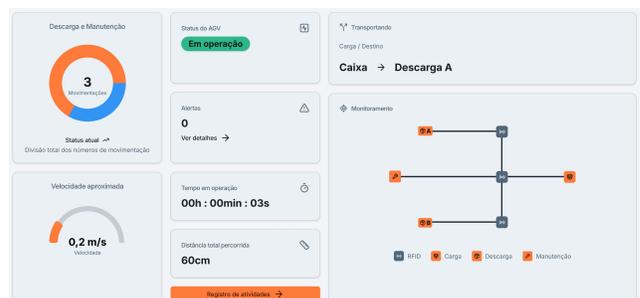


Figura 10. Componentes da Visão Geral do aplicativo *web* [31].

Além disso, a Figura 11 apresenta os principais componentes da tela de Gestão de Transporte, onde são exibidas informações detalhadas sobre o transporte em andamento. Essa interface também oferece funcionalidades como alternar para o modo de manutenção, cadastrar uma nova carga ou iniciar e cancelar transportes. No modo de manutenção, o operador pode assumir o controle total do AGV utilizando um rádio controle, permitindo movimentá-lo em casos de problemas ou direcioná-lo à área de manutenção.



Figura 11. Componentes da Gestão de Transporte do aplicativo *web* [31].

Para iniciar um transporte, o operador utiliza a interface ilustrada na Figura 12. Nela, é possível selecionar as informações da carga previamente cadastrada, como o tipo, as dimensões e o destino. Após a confirmação pelo botão "Iniciar", a solicitação é enviada ao AGV, que dá início ao transporte.



Figura 12. Janela de Seleção de Transporte do aplicativo *web* [31].

Por fim, o aplicativo conta com um recurso de Registro de Atividades, projetado para exibir os componentes que permitem monitorar as ações realizadas pelo operador e pelo AGV. Esse recurso facilita a identificação e resolução de problemas, além de manter um histórico das operações, garantindo maior controle sobre o sistema.

VI. CONCLUSÃO

Este trabalho apresentou a construção de um protótipo de sistema no contexto da Indústria 4.0, envolvendo o uso de sensores *Radio-Frequency Identification* (RFID) e ultrassônicos, em conformidade com normas específicas para operação segura, como a *Verein Deutscher Ingenieure 2510* (VDI 2510) para sistemas de *Automatized Guided Vehicles* (AGV) e a Norma Regulamentadora No. 26 (NR-26).

Adicionalmente, a integração baseada na Internet das Coisas foi realizada por meio do desenvolvimento de um aplicativo *web* para controle e monitoramento do AGV ao longo do trajeto.

A utilização da metodologia Kanban facilitou o desenvolvimento, pois incentiva a prototipagem, importante devido às diversas modificações pelas quais o veículo passou até atingir o resultado final. Além disso, essa metodologia permitiu manter a organização clara e objetiva por meio do quadro Kanban.

Os resultados mostraram que o sistema atende às operações propostas. O uso do sensor RFID trouxe consistência ao realizar o trajeto, mas limitou a precisão em curvas e a capacidade de carga. O sensor ultrassônico mostrou-se eficaz na detecção de obstáculos. No entanto, a ausência de um sensor de refletância restringiu o uso do AGV a trajetos complexos e longos.

O aplicativo *web* contribuiu para a organização e monitoramento das operações, oferecendo recursos como Visão Geral, Gestão de Transporte e Registro de Atividades, evidenciando a velocidade de comunicação entre os diferentes sistemas.

Para trabalhos futuros, propõe-se a implementação de um sensor de refletância adequado ao contexto. Além disso, pode-se aprimorar o monitoramento da posição do AGV, permitindo o acompanhamento da posição exata do veículo ao longo do trajeto. Por fim, a integração com o *software* específico de manufatura da empresa que utilizará o AGV poderá facilitar o controle de estoque e a identificação dos operadores por meio de um sistema de recursos humanos.

REFERÊNCIAS

- [1] Bitkom, VDMA e ZVEI. *Implementation Strategy Industrie 4.0: Report on the Results of the Industrie 4.0 Platform*. Disponível em: https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2016/januar/Implementation_Strategy_Industrie_4.0_-_Report_on_the_results_of_Industrie_4.0_Platform/Implementation-Strategy-Industrie-40-ENG.pdf. Acessado em: 01 de maio de 2024.
- [2] A. Ustundag e E. Cevikkan. *Industry 4.0: Managing the Digital Transformation*. Springer, 2018. ISBN: 978-3-319-57869-9. DOI: 10.1007/978-3-319-57870-5.

- [3] M. P. Groover. *Automation, Production Systems, and Computer-Integrated Manufacturing*. Pearson Custom Library. Pearson, 2014. ISBN: 9781292025926.
- [4] C. Liu, M. Fan e R. Song. “A Novel Intelligent and Agile Warehouse System for Energy Meter Storage”. Em: *2013 IEEE Third International Conference on Information Science and Technology (ICIST)*. IEEE, 2013, pp. 129–133.
- [5] J. N. Pires. *Automação e Controlo Industrial - Indústria 4.0*. Lidel, 2019. ISBN: 978-989-752-226-1.
- [6] M. P. Groover. *Fundamentals of Modern Manufacturing: Materials, Processes, and Systems*. John Wiley & Sons, 2010. ISBN: 9780470467008.
- [7] H. Song et al. *Cyber-Physical Systems: Foundations, Principles and Applications*. Intelligent Data-Centric Systems. Elsevier Science, 2016. ISBN: 9780128038741.
- [8] IBM. *O que é Internet das Coisas (IoT)?* Disponível em: <https://www.ibm.com/br-pt/topics/internet-of-things>. Acessado em: 10 de abril de 2024.
- [9] K. Schwab. *The Fourth Industrial Revolution*. Portfolio, 2017. ISBN: 9780241300756.
- [10] G. Ullrich e T. Albrecht. *Automated Guided Vehicle Systems: A Guide with Practical Applications About the Technology for Planning*. Springer Fachmedien Wiesbaden, 2022. ISBN: 9783658353872.
- [11] Verein Deutscher Ingenieure. *Specification for the Operation of Automated Guided Vehicles*. Technical Specification VDI 2510. Alemanha: Verein Deutscher Ingenieure, 2021.
- [12] Ministério do Trabalho e Emprego. *Norma Regulamentadora No. 26 (NR-26)*. Norma Especial 26.1.5.3 Amarelo, 126.004-9/I2. Brasil: Ministério do Trabalho e Emprego, 2012.
- [13] E. Upton e G. Halfacree. *Raspberry Pi User Guide*. Wiley, 2014. ISBN: 9781118921661.
- [14] S. V. dos Santos. *RFID: Conceitos, Implementação e Desempenho com Baixo Custo Computacional*. Editora Dialética, 2022. ISBN: 9786525232386.
- [15] Pololu Corporation. *QTR-8A and QTR-8RC Reflectance Sensor Array User’s Guide*. Disponível em: <https://www.pololu.com/docs/pdf/0J12/QTR-8x.pdf>. Acessado em: 04 de abril de 2024.
- [16] K. Schwab. *The Fourth Industrial Revolution*. Portfolio, 2017. ISBN: 9780241300756.
- [17] Python Software Foundation. *What is Python? Executive Summary*. Disponível em: <https://www.python.org/doc/essays/blurb>. Acessado em: 06 de abril de 2024.
- [18] D. George. *MicroPython*. Disponível em: <https://micropython.org>. Acessado em: 06 de abril de 2024.
- [19] Mozilla Corporation. *HTTP*. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/HTTP>. Acessado em: 17 de junho de 2024.
- [20] HiveMQ. *MQTT Essentials: The Ultimate Guide to the MQTT Protocol for IoT Messaging*. Disponível em: <https://www.hivemq.com/resources/download-mqtt-ebook>. Acessado em: 17 de junho de 2024.
- [21] Facebook Open Source. *What is the Virtual DOM?* Disponível em: <https://legacy.reactjs.org/docs/faq-internals.html>. Acessado em: 29 de abril de 2024.
- [22] A. Banks e E. Porcello. *Learning React: Functional Web Development with React and Redux*. O’Reilly Media, 2017.
- [23] Vercel. *What is Next.js?* Disponível em: <https://nextjs.org/docs>. Acessado em: 05 de abril de 2024.
- [24] Michael Goodwin. *What is an API?* Disponível em: <https://www.ibm.com/topics/api>. Acessado em: 13 de novembro de 2024.
- [25] Vercel. *tRPC - Introduction*. Disponível em: <https://trpc.io/docs>. Acessado em: 13 de novembro de 2024.
- [26] J. Mehami, M. Nawi e R. Y. Zhong. “Smart Automated Guided Vehicles for Manufacturing in the Context of Industry 4.0”. Em: *Procedia Manufacturing* 26 (2018). 46th SME North American Manufacturing Research Conference, NAMRC 46, Texas, USA, pp. 1077–1086. ISSN: 2351-9789. DOI: 10.1016/j.promfg.2018.07.144.
- [27] H. Sasamoto et al. “Modeling and Prototype Implementation of an Automated Guided Vehicle for Smart Factories”. Em: *2021 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)*. 2021, pp. 1–6. DOI: 10.1109/ICMLANT53170.2021.9690543.
- [28] K. Ananthi et al. “IoT Based AGV System for Medicare”. Em: *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)*. 2022, pp. 410–414. DOI: 10.1109/ICESC54411.2022.9885709.
- [29] I. Sommerville. *Engenharia de Software*. Pearson Prentice Hall, 2011. ISBN: 9788579361081.
- [30] R. S. Pressman et al. *Engenharia de Software*. AMGH, 2021. ISBN: 9786558040101.
- [31] M. P. Braga. *Desenvolvimento de um Protótipo de Veículo Automatizado para Automação Logística na Indústria 4.0*. Disponível em: <https://github.com/mauricioprbr/tfg-agv-sistema>. Acessado em: 17 de junho de 2024.
- [32] H. H. Hamzah et al. “3D Printable Conductive Materials for the Fabrication of Electrochemical Sensors: A Mini Review”. Em: *Electrochemistry Communications* 96 (2018), pp. 27–31. ISSN: 1388-2481. DOI: 10.1016/j.elecom.2018.09.006.