



Instituto Superior de Engenharia

Politécnico de Coimbra

DEPARTAMENTO DE ENGENHARIA
ELETROTÉCNICA

Desenvolvimento de um software de supervisão e monitorização da frota de Automated Guided Vehicles

Dissertação para a obtenção do grau de Mestre em Engenharia
Eletrotécnica

Especialização em Automação e Comunicações em Sistemas
Industriais

Autor

David Francisco Carvalho Lopes

Orientadores

Professor Doutor Nuno Miguel Fonseca Ferreira

Professor Coordenador Principal

Professor Doutor Victor Daniel Neto dos Santos

Professor Coordenador



INSTITUTO POLITÉCNICO
DE COIMBRA

INSTITUTO SUPERIOR
DE ENGENHARIA
DE COIMBRA

Coimbra, 12/2024

AGRADECIMENTOS

No final deste projeto, gostaria de expressar a minha mais sincera gratidão a todos aqueles que contribuíram para o seu desenvolvimento.

Em primeiro lugar, agradeço ao Professor Nuno Miguel Fonseca Ferreira e ao Professor Victor Daniel Neto dos Santos, orientadores do ISEC, pelo apoio e pela orientação ao longo de todo o projeto, cujo contributo foi fundamental para o sucesso deste trabalho.

Um agradecimento especial à equipa do laboratório da Robocorp presente no ISEC, que esteve sempre disponível para me ajudar, em particular ao André Gonçalves, ao Francisco Cunha, ao Tiago Pereira e ao Fernando Lopes, cuja colaboração e conhecimentos foram essenciais.

Agradeço também à equipa da Europneumaq, empresa líder do PPS em que este projeto se insere, e, em especial, ao Engenheiro Rui Bessa, pelo apoio e confiança que depositaram em mim.

Não posso deixar de expressar a minha profunda gratidão à minha namorada, pelo seu apoio constante, paciência e dedicação.

Por fim, gostaria também de agradecer à minha família por todas as palavras de incentivo, pelo carinho e por tornarem possível a continuidade da minha formação, apoiando-me incondicionalmente em cada etapa desta jornada, independentemente das minhas decisões.

RESUMO

O projeto "GreenAuto: Inovação Verde para a Indústria Automóvel" visa fortalecer a indústria automóvel, promovendo a produção de veículos com baixas emissões otimizando os respetivos processos produtivos. No âmbito deste projeto existe uma Work Package (WP) dedicada à logística automatizada para a indústria automóvel, na qual se pretende melhorar a eficiência operacional da unidade fabril da Stellantis, localizada em Mangualde. Para esse efeito, estão a ser desenvolvidas soluções para os sistemas de supervisão e de navegação dos Veículos Guiados Automaticamente (AGVs) existentes.

Atualmente, a empresa não dispõe de um sistema de gestão de frotas funcional que permita monitorizar a localização e o estado de funcionamento dos AGVs. Esta situação é agravada pelo facto de serem utilizados vários modelos e marcas de AGVs, tornando difícil a recolha e a padronização dos dados.

Desta forma, o projeto/dissertação descrito neste relatório tem como objetivo desenvolver e implementar uma ferramenta eficiente para monitorizar e supervisionar todos os AGVs presentes no chão de fábrica. A aplicação processa os dados recolhidos dos AGVs e armazena-os numa base de dados central.

Além disso, esta aplicação permite localizar os AGVs em tempo real e identificar o seu estado operacional através de um sinóptico de cores intuitivo, bem como aceder às suas informações mais relevantes.

O sistema regista o histórico de movimentações, facilitando a manutenção preventiva e a otimização das rotas. Para assegurar uma comunicação contínua e fiável, será instalada uma rede *mesh* com múltiplos pontos de acesso, capaz de cobrir toda a área da fábrica, garantindo a disponibilidade interrupta da comunicação entre os AGVs e o servidor.

O projeto propõe também uma solução escalável que permite a futura integração de novos modelos de AGVs e robôs móveis, como os Robôs Móveis Autónomos (AMRs). Esta abordagem visa aumentar a eficiência e a competitividade da fábrica face às evoluções tecnológicas do setor automóvel.

Palavras-Chave: AGVs, Base de dados, Monitorização, Pontos de Acesso, Supervisão.

ABSTRACT

The project "GreenAuto: Green Innovation for the Automotive Industry" aims to strengthen the automotive industry by promoting the production of low-emission vehicles and optimizing their production processes. Within the scope of this project, there is a dedicated Work Package (WP) focused on automated logistics for the automotive industry, with the goal of improving the operational efficiency of the Stellantis manufacturing facility located in Mangualde.

To achieve this, solutions are being developed for the supervision and navigation systems of the existing Automated Guided Vehicles (AGVs). Currently, the company lacks a functional fleet management system capable of monitoring the location and operational status of the AGVs. This challenge is further exacerbated using various AGV models and brands, making it difficult to collect and standardize data.

Therefore, the project described in this report aims to develop and implement an efficient tool to monitor and supervise all AGVs present on the factory floor. The application processes data collected from the AGVs and stores it in a central database.

Furthermore, this application enables real-time localization of the AGVs and identifies their operational status using an intuitive color-coded synoptic system, while also providing access to their most relevant information. The system records movement history, facilitating preventive maintenance and route optimization.

To ensure continuous and reliable communication, a mesh network with multiple access points will be installed, capable of covering the entire factory area and ensuring uninterrupted communication between the AGVs and the server.

The project also proposes a scalable solution that allows for the future integration of new AGV models and mobile robots, such as Autonomous Mobile Robots (AMRs). This approach aims to enhance the factory's efficiency and competitiveness in response to technological advancements in the automotive sector.

Keywords: Access Points, AGVs, Database, Monitoring, Supervision.

ÍNDICE

Conteúdo

AGRADECIMENTOS	i
RESUMO	ii
ABSTRACT	iii
ÍNDICE	iv
ÍNDICE DE FIGURAS	vii
ÍNDICE DE QUADROS	xi
SIMBOLOGIA E ABREVIATURAS	xii
INTRODUÇÃO	1
Enquadramento	1
Problemática e requisitos da solução a desenvolver	3
Objetivos	4
Estrutura do documento	5
ESTADO DA ARTE	7
2.1 – Indústria 4.0	7
2.1.1 – Vantagens e Desvantagens	8
2.1.2 – Desafios e riscos associados	8
2.1.3 – Planos futuros	9
2.2 – Robótica Móvel	10
2.2.1 – Evolução da robótica móvel	11
2.3 – Automated Guided Vehicles - AGVs	15
2.3.1 – Vantagens e desvantagens	16
2.3.2 – Segurança	16
2.3.3 – Movimentação	20
2.3.4 – Rotas	22
2.3.5 – Carregamento dos AGVs	23
2.3.6 – Navegação	24
2.3.7 – Localização	25
2.3.8 – Monitorização	26
2.4 – HMI	27
2.4.1 – História das HMIs	28
2.4.2 – Vantagens e desvantagens	29
2.4.3 – Tipos de HMI	30
2.4.5 – Aplicações	31
2.4.6 – Avanços futuros nas HMIs	32

2.5 – Sistema flexível de manufatura FMS	33
2.5.1 – Aparecimento do FMS	34
2.5.2 – Vantagens e desvantagens	34
2.5.3 – Utilização do FMS em conjunto com a HMI	34
DESENVOLVIMENTO	37
3.1 – Hardware utilizado	38
3.1.1 – Servidor	38
3.1.2 – Router Wireless.....	38
3.1.3 – Access points.....	39
3.1.4 – Datalogger	40
3.1.5 – AGVs	41
3.1.6 – Banda magnética e tags RFID.....	42
3.2 – Softwares utilizados	43
3.2.1 – Software dos APs da marca Grandstream	43
3.2.2 – AutoCAD	44
3.2.3 – Aplicação da ASTI.....	45
3.2.4 – SIGAT.....	45
3.2.5 – SIGAT MultiAGV	46
3.2.6 – Software dos APs dos AGVs (Acksys)	46
3.2.7 – Xampp.....	47
3.2.8 – PHPMyAdmin	47
3.2.9 – SQL Server Management Studio	48
3.2.10 – MySQL Workbench	48
3.2.11 – Linguagem Python.....	48
3.2.12 – Linguagem PHP	49
3.2.13 – Linguagem JavaScript.....	49
3.2.14 – Linguagem CSS	50
3.2.15 – Linguagem HTML.....	51
3.3 – Recolha da informação das rotas e tags	51
3.4 – Criação de uma rede com APs.....	52
3.5 – Testes com o AGV	54
3.6 – Recolha de dados dos AGVs	55
3.6.1 – Versão mais recente dos AGVs	55
3.6.2 – Versão mais antiga dos AGVs	59
3.7 – Definição da base de dados para o software em desenvolvimento	62
3.8 – Desenvolvimento da plataforma de monitorização e de supervisão.....	65

3.8.1 – Página de Login	65
3.8.2 – Edição de rotas e tags	67
3.8.3 – Posição atual dos AGVs.....	69
3.8.4 – Supervisão dos AGVs	71
3.9 – Estado final do software de supervisão e monitorização	74
CONCLUSÃO	79
4.1 – Objetivos alcançados e análise de resultados	79
4.2 – Trabalho futuro.....	80
REFERÊNCIAS BIBLIOGRÁFICAS	81
ANEXOS	89
Anexo A – Index.php.....	89
Anexo B – Pagina.html	90
Anexo C – Topnav.html	91
Anexo D – Logout.php.....	92
Anexo E – Sistema.php	92
Anexo F – Script.js.....	98
Anexo G – DB_Projeto_Tese.sql.....	99

ÍNDICE DE FIGURAS

Figura 1 – Logotipo da Agenda Green Auto [1]	1
Figura 2 – Divisão dos PPS do projeto.....	1
Figura 3 – Evolução da indústria [6]	7
Figura 4 – Indústria 4.0 [10].....	8
Figura 5 – Robótica nos dias de hoje [13]	9
Figura 6 – Os AGVs utilizam uma banda magnética como guia [19]	10
Figura 7 – Os AMRs não precisam de guia [20].....	10
Figura 8 – AGVs da Barret Electronics (versão 1 e 2) [22].....	11
Figura 9 – Primeiro AGV de carga unitária [23].....	11
Figura 10 – AGV a utilizar uma banda magnética para a navegação automática [24]	12
Figura 11 – Primeiro AGV a utilizar um sistema de lasers para a navegação automática [25]	12
Figura 12 – Diferença entre a deteção de um obstáculo entre A e B [26]	13
Figura 13 – AGVs utilizados numa das fábricas da TESLA [27].....	14
Figura 14 – Robôs móveis utilizados num dos armazéns da Amazon [28]	14
Figura 15 – Gama de AGVs da ASTI/ABB [32]	15
Figura 16 – Ângulos de deteção.....	17
Figura 17 – Luz vermelha, indicando anomalia ou modo de emergência	17
Figura 18 – Luz amarela, indicando modo autónomo	17
Figura 19 – Luz azul, indicando modo manual	17
Figura 20 – Comando para modo manual.....	18
Figura 21 – Botoneira.....	18
Figura 22 – Zona de segurança [35]	18
Figura 23 – Laser de segurança SICK [36]	18
Figura 24 – Câmara Alvium U [37]	19
Figura 25 – Câmara SICK SafeVisionary 2 [38]	19
Figura 26 – Sensor ultrassónico [39].....	19
Figura 27 – Zona de ultrassons [40].....	19
Figura 28 – AGV com para-choques de fábrica	19
Figura 29 – AGV com para-choques adicionado posteriormente [41].....	19
Figura 30 – Roda Fixa [43].....	21
Figura 31 – Roda Livre [44].....	21
Figura 32 – Roda Motriz [45].....	21
Figura 33 – Roda Direção/Motriz [46].....	21
Figura 34 – Roda Omnidirecional [47].....	21
Figura 35 – Unidirecional.....	22
Figura 36 – Bidirecional.....	22
Figura 37 – Omnidirecional	22
Figura 38 – Rota com linha única.....	22
Figura 39 – Rota em loop	22
Figura 40 – Rotas com vários loops	23

Figura 41 – Trafic BOX.....	23
Figura 42 – Principais tipos de navegação em AGVS [52]	25
Figura 43 – Lógica das HMIs [57].....	27
Figura 44 – Primeiro computador ENIAC e os cartões de dados [59]	28
Figura 45 – Xerox Alto [61].....	28
Figura 46 – Um operador a utilizar uma HMI [62].....	30
Figura 47 – Diferentes tipos de HMI da marca Siemens [66]	31
Figura 48 – Tipos de aplicações de HMIs [69]	32
Figura 49 – HMI com VR e AR em desenvolvimento pela BMW [72]	32
Figura 50 – Exemplo de FMS.....	33
Figura 51 – Servidor utilizado no projeto [75].....	38
Figura 52 – Router TP-Link Router Wireless utilizado na rede local [76]	39
Figura 53 – Access Points da Grandstream	39
Figura 54 – Access Points da Acksys [77]	40
Figura 55 – Datalogger existente em cada um dos AGVs [78].....	40
Figura 56 – ASTI BidiBot [79].....	41
Figura 57 – ASTI EasyBot [80]	41
Figura 58 – ASTI TriBot [81].....	41
Figura 59 – Material utilizado (Banda magnética e Tags RFID) [82]	43
Figura 60 – Página inicial do software dos APs	44
Figura 61 – Aplicação AutoCAD com o exemplo de uma rota.....	44
Figura 62 – Aplicação Web da ASTI	45
Figura 63 – Aplicação da ASTI (SIGAT).....	45
Figura 64 – SIGAT MultiAGV	46
Figura 65 – Software dos APs dos AGVs (Acksys)	46
Figura 66 – Logotipo do software Xampp [83].....	47
Figura 67 – Página inicial do PHPMyAdmin	47
Figura 68 – SQL Server Management Studio.....	48
Figura 69 – Logotipo do MySQL Workbench [84].....	48
Figura 70 – Logotipo do Python [85]	49
Figura 71 – Logotipo do PHP [86]	49
Figura 72 – Logotipo do JavaScript [87].....	50
Figura 73 – Logotipo do CSS [88]	50
Figura 74 – Logotipo do HTML [89].....	51
Figura 75 – Excerto de uma rota retirada com as respetivas tags	52
Figura 76 – Configuração do Router utilizado para a criação da rede local	52
Figura 77 – Configuração dos APs utilizados para a criação de uma rede mesh	53
Figura 78 – Detalhes dos equipamentos conectados à rede local	53
Figura 79 – AGV com mais abundância na Stellantis [90]	54
Figura 80 – Várias opções de programação na aplicação SIGAT	54
Figura 81 – Exemplo de rota utilizada para testes	55
Figura 82 – Configuração do IP correto para aceder ao AGV	56
Figura 83 – Página inicial dos dataloggers da ASTI.....	56

Figura 84 – Janela do Real-Time Log da página da ASTI.....	57
Figura 85 – Dados recolhidos do AGV pelo datalogger	57
Figura 86 – Dados retirados da página da ASTI processados	58
Figura 87 – Informação dos AGV pela aplicação da ASTI	59
Figura 88 – Base de dados na aplicação da ASTI	59
Figura 89 – Interface do programa SIGAT MultiAGV	60
Figura 90 – Tabelas existentes na base de dados do SIGAT <i>MultiAGV</i>	61
Figura 91 – <i>Query</i> principal com a informação relevante resumida	62
Figura 92 – Base de dados para o software em desenvolvimento	62
Figura 93 – Excerto um da base de dados desenvolvida	63
Figura 94 – Excerto dois da base de dados desenvolvida	63
Figura 95 – Tabelas com programação dos AGVs.....	64
Figura 96 – Exemplo da informação das tabelas	64
Figura 97 – Página de login da página web	65
Figura 98 – Página web versão 1	67
Figura 99 – Lógica da mudança de rota na página web.....	68
Figura 100 – Lógica de guardar tags na página web	69
Figura 101 – Lógica para a posição do AGV na página web	70
Figura 102 – Exemplo da representação da posição do AGV na página web	71
Figura 103 – Rota do ISEC correspondente ao ID 12 na base de dados.....	73
Figura 104 – Exemplo das informações relativas ao AGV selecionado	73
Figura 105 – AGV vermelho.....	74
Figura 106 – AGV amarelo.....	74
Figura 107 – AGV Azul.....	74
Figura 108 – AGV verde.....	74
Figura 109 – Estado final da página de login	75
Figura 110 – Estado final da página “Home”	75
Figura 111 – Estado final da página de “Edição de Rotas e Tags”	76
Figura 112 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Running”	76
Figura 113 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Alert”	77
Figura 114 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Emergency”	77

ÍNDICE DE QUADROS

Tabela 1 – Tipos de Rodas em AGVs	21
Tabela 2 – Principais características dos AGVs.....	42
Tabela 3 – Tabela do trajeto programado da rota do ISEC.....	72
Tabela 4 – Sinóptico das cores do estado dos AGVs	74

SIMBOLOGIA E ABREVIATURAS

AGV	Automated Guided Vehicle
AMR	Autonomous Mobile Robot
AP	Access Point
AR	Augmented Reality
ASTI	Automatismos y Sistemas de Transporte Interno
BE-LCV	Battery Electric Light Commercial Vehicle
CAD	Computer-Aided Design
CSS	Cascading Style Sheets
DOM	Document Object Model
ENIAC	Electronic Numerical Integrator and Computer
FMS	Flexible Manufacturing System
GPS	Global Positioning System
GUI	Graphical User Interface
HMI	Human-Machine Interface
HTML	HyperText Markup Language
IA	Inteligência Artificial
IoT	Internet of Things
ISEC	Instituto Superior de Engenharia de Coimbra
JS	JavaScript
LiDAR	Light Detection And Ranging
M2M	Machine to Machine
PPS	Produtos e Processos Sustentáveis
PPR	Plano de Recuperação e Resiliência
QR	Quick Response
RFID	Radio Frequency Identification
SLAM	Simultaneous Localization And Mapping
VR	Virtual Reality
WP	Work Package

INTRODUÇÃO

Enquadramento

A indústria automóvel está a passar por uma transformação significativa devido à necessidade urgente de reduzir a sua pegada ecológica. Dado o seu impacto significativo na economia, foi lançada a Agenda GreenAuto, cujo logotipo está representado na Figura 1, para posicionar o setor automóvel nacional como líder na produção de veículos comerciais ligeiros de baixas emissões.

A Agenda Mobilizadora para a Inovação Empresarial, intitulada "GreenAuto: Inovação Verde para a Indústria Automóvel", é um projeto cofinanciado pelo Plano de Recuperação e Resiliência (PRR), com o objetivo de fortalecer a posição da indústria automóvel nacional na cadeia de valor dos veículos de baixas emissões [1].



Figura 1 – Logotipo da Agenda Green Auto [1]

Para alcançar este objetivo serão desenvolvidos 18 produtos e processos inovadores, digitais e sustentáveis (PPS). O projeto está estruturado em dois eixos de atuação, representados na Figura 2 Green Vehicle e Green Factory [2].



Figura 2 – Divisão dos PPS do projeto

O PPS1 representa o produto principal proposto pela Agenda GreenAuto, que compreende a produção de um veículo comercial ligeiro de baixas emissões, designado em inglês por Battery Electric - Light Commercial Vehicle (BE-LCV).

O eixo de atuação Green Vehicle | Produto engloba os seguintes PPS:

- PPS2: Materiais funcionais, inteligentes e digitais;
- PPS3: Têxteis técnicos;
- PPS4: Espumas sustentáveis;
- PPS5: Módulos de airbag;
- PPS6: Volante HMI;
- PPS7: Sistemas de fecho mais leves.

Por outro lado, o eixo de atuação Green Factory | Processo inclui os seguintes PPS:

- PPS8: Sistemas automatizados de montagem de obturadores;
- PPS9: Equipamentos de pintura revestidos com materiais super-hidrofóbicos e ecológicos;
- PPS10: Sistemas de redes neurais para robôs;
- PPS11: Novo método para geração de silício de alta pureza;
- PPS12: Sistemas de controlo automatizado para manipulação de alta precisão;
- PPS13: Sistemas de robótica colaborativa para a linha de montagem;
- PPS14: Sistema de controlo de qualidade preditivo para soldadura;
- PPS15: Sistema de orientação por realidade virtual e artificial;
- PPS16: Sistema de drones para controlo de qualidade;
- PPS17: Sistema de supervisão de frota dos *Automated Guided Vehicle* (AGVs);
- PPS18: sistema de navegação 3D para robôs móveis.

Esta Agenda é constituída por dez Work Packages (WP), denominados da seguinte forma:

- WP 1: Veículos comerciais verdes inovadores;
- WP 2: Novos conceitos para o interior do automóvel;
- WP 3: Desenvolvimento de têxteis com propriedades de captação de energia fotovoltaica;
- WP 4: Desenvolvimento de espumas com alta incorporação de resíduos;
- WP 5: Desenvolvimento e industrialização de novos módulos de Airbags;

- WP 6: Desenvolvimento e Industrialização de volantes com HMI;
- WP 7: Sistemas de fecho sem juntas, compactos e mais leves;
- WP 8: Sistemas robóticos e tecnologias de fabricação para a indústria automóvel;
- WP 9: Sistema de controlo de qualidade para a indústria automóvel;
- WP 10: Logística automatizada para a indústria automóvel.

O WP 1 foca-se na industrialização de um BE-LCV, enquanto os WP 2 a 7 centram-se no desenvolvimento de componentes e sistemas inovadores para veículos elétricos. Os WP 8 a 10 centram-se no desenvolvimento de tecnologias de produção mais eficientes e produtivas [3].

O trabalho no âmbito deste projeto está centrado na WP 10 - Logística automatizada para a indústria automóvel. Este WP foca-se no desenvolvimento de soluções para os sistemas de supervisão dos AGVs, propondo-se ainda o desenvolvimento de um sistema de navegação mais robusto para contextos produtivos complexos presentes na indústria.

Este WP é composto por dois PPS: o PPS17 e o PPS18, sendo que o foco do projeto desenvolvido será no PPS17 - Sistema de supervisão de frotas de AGVs. Este PPS propõe o desenvolvimento de módulos para os sistemas de gestão de AGVs, integrando características como:

- Sistema de visão artificial 3D para maximizar a autonomia e mapeamento do ambiente envolvente;
- Sistema de supervisão da frota de AGV focada na otimização do consumo de energia.

Problemática e requisitos da solução a desenvolver

Na fábrica da Stellantis, existe uma grande diversidade de AGVs, de diferentes modelos, que são utilizados para distribuição de material para os diferentes postos da linha de produção, sendo estes guiados por fitas magnéticas coladas ao chão, sobre as quais se encontram tags com diferentes comandos para serem executados. Os AGVs estão distribuídos pelas linhas de montagem e produção, abrangendo tanto os modelos mais antigos como os de última geração. O desafio surge com a diferença entre os softwares dos diferentes modelos existentes, pois é necessário garantir que todos os AGVs comuniquem com um servidor de uma forma padrão para assegurar o bom funcionamento e eficiência. Assim, para realizar este feito, torna-se essencial implementar um software de supervisão e monitorização para a frota de AGVs presentes na fábrica.

Objetivos

Para que a integração do software de supervisão e monitorização da frota dos vários modelos de AGVs decorra sem problemas, é essencial desenvolver uma solução que permita comunicar de forma eficiente com todos os dispositivos, apesar das suas diferenças. Isto implica o desenvolvimento de uma solução que extraia a informação necessária de cada tipo de AGV. Esta informação que vai ser retirada terá de ser posteriormente processada para converter todos os dados e comandos necessários para um formato padrão, que será utilizado pelo software de supervisão e monitorização. Estas soluções atuarão como uma ponte entre os AGVs e o servidor central de supervisão e monitorização, permitindo que este interaja de forma uniforme com todos os modelos.

A primeira tarefa passa pelo levantamento detalhado das características técnicas e dos protocolos de comunicação de cada modelo diferente de AGVs. Com base nesta informação, será possível desenvolver a solução necessária que irá adaptar os dados de todos os AGVs para um formato padrão para que estes possam ser utilizados pelo software de supervisão, o qual é composto essencialmente por uma Interface Homem Máquina também conhecida por *Human Machine Interface* (HMI) e um sistema flexível de manufatura designado em inglês por *Flexible Manufacturing System* (FMS). Foi também realizado um levantamento das rotas e das tags existentes na fábrica de forma a estruturar corretamente o desenvolvimento do projeto. Refira-se que as tags inseridas nas rotas existentes são responsáveis por guiar os AGVs e identificar a sua passagem por pontos predefinidos.

Na próxima tarefa, será necessário criar uma rede *mesh* com vários *Access Points* (APs) que abranjam toda a área útil da fábrica. Desta forma, à medida que os AGVs se movimentam pela fábrica, estes vão estar sempre ligados ao AP com a conexão mais forte, garantindo assim a disponibilidade contínua da comunicação com o servidor de supervisão.

Com os AGVs já interligados ao servidor e com uma solução para a recolha e processamento dos dados, será iniciado o desenvolvimento da página web que funcionará como HMI, na qual será possível executar determinadas tarefas e consultar várias informações.

O objetivo final deste projeto é garantir que o software de supervisão e monitorização funcione corretamente para todos os tipos de AGVs. É também crucial que o sistema HMI/FMS seja projetado com uma estrutura flexível, permitindo uma adaptação contínua às evoluções tecnológicas, para que no futuro, caso sejam adicionados AGVs mais recentes, não surjam quaisquer problemas. Esta flexibilidade garantirá uma integração suave de novos modelos de AGVs no futuro, sem interrupções significativas ou custos excessivos. Ao adotar esta abordagem, a empresa será capaz de otimizar ainda mais os processos de produção, promovendo uma melhoria contínua o que resultará numa produção mais eficiente, adaptável e competitiva num mercado em constante mudança.

Estrutura do documento

Este documento encontra-se dividido em quatro capítulos.

No Capítulo 1, “Introdução”, é realizado o enquadramento da problemática, quais os objetivos propostos e qual a estrutura do documento.

O Capítulo 2, referente ao “Estado da Arte”, tem o objetivo de apresentar uma breve atualização sobre estado atual da tecnologia, mais precisamente sobre os tópicos da indústria 4.0, da robótica móvel, dos AGVs, das HMIs e dos FMS. Em cada um destes tópicos, é realizada uma descrição que inclui a explicação dos conceitos, os primeiros aparecimentos e a evolução, as vantagens e desvantagens de cada tipo de tecnologia, bem como os objetivos futuros e implicações da sua implementação.

O Capítulo 3, que constitui a secção mais extensa deste relatório do projeto, está estruturado em diversos tópicos principais, incluindo: os dispositivos utilizados, os programas de software utilizados, a recolha de informações das rotas e tags, a criação de uma rede com APs, os testes realizados com AGVs, a recolha de dados dos AGVs, o desenvolvimento de uma base de dados para o software em desenvolvimento, o desenvolvimento do software/plataforma web de monitorização e supervisão, e a demonstração do estado final do software de monitorização e supervisão. Cada um destes tópicos é explorado através de subcapítulos que abordam temas específicos, mas todos eles orientados para a concretização do produto final.

O Capítulo 4, intitulado “Conclusão”, sintetiza e encerra todo o trabalho desenvolvido neste projeto. Este capítulo está organizado em dois tópicos principais: os objetivos alcançados, onde se destacam os resultados obtidos, e a análise de resultados, que oferece uma reflexão sobre os desenvolvimentos alcançados. Adicionalmente, é apresentado o trabalho futuro, que fornece uma breve descrição das propostas para a continuidade deste projeto, com foco na evolução e aperfeiçoamento do software desenvolvido.

ESTADO DA ARTE

2.1 – Indústria 4.0

A Indústria 4.0 ou a quarta revolução industrial, representa uma alteração na forma de como os produtos são desenvolvidos, produzidos e distribuídos. Esta alteração é impulsionada pela incorporação de tecnologias, como a automação, a análise de dados e a interligação entre diferentes máquinas [4]

A industrialização começou no final do século XVIII, como se pode observar na Figura 3, com a Indústria 1.0, marcada pela introdução da energia hídrica e a de vapor. No século XIX, surgiu a Indústria 2.0, com a produção em massa a ser alimentada pela energia elétrica. No século XX, ocorreu a terceira revolução industrial, caracterizada pela automação industrial que visava aumentar a eficiência e produtividade da indústria.

A Indústria 4.0 surge como um avanço natural desta trajetória, com base na Indústria 3.0, nomeadamente o aparecimento da Internet e a digitalização dos processos. A capacidade de integrar máquinas, sistemas e pessoas numa única rede lógica e inteligente distingue a Indústria 4.0 das suas antecessoras, promovendo uma produção mais flexível e eficiente [5].



Figura 3 – Evolução da indústria [6]

Além disso, a interligação digital permite a comunicação e cooperação em tempo real entre os operadores, máquinas e produtos, otimizando o fluxo de trabalho e aumentando significativamente a eficiência. A Indústria 4.0 está também a impulsionar inovações em áreas como a inteligência artificial, Internet das Coisas (IoT), a análise de dados e a realidade aumentada, entre outras [7].

2.1.1 – Vantagens e Desvantagens

A transição para a Indústria 4.0 apresenta múltiplas vantagens, entre as quais se destacam as seguintes.

- **Eficiência operacional:** a integração de sistemas de automação e a digitalização dos processos contribuem para uma produção mais eficaz, minimizando os custos operacionais e maximizando a produtividade.
- **Flexibilidade:** a capacidade de adaptação rápida às alterações do mercado é significativamente maior, graças a uma maior flexibilidade nos processos de produção.
- **Melhoria na qualidade do produto:** a implementação de tecnologias avançadas permite um controlo mais rigoroso sobre a qualidade dos produtos, reduzindo a probabilidade de produtos defeituosos.
- **Sustentabilidade:** a Indústria 4.0 promove operações mais sustentáveis através da otimização de recursos na produção, Figura 4.

No entanto, a transição para a Indústria 4.0 também apresenta desvantagens, como sendo a principal delas a sua complexidade e os custos elevados que acarreta, uma vez que a transição para esta nova era industrial exige investimentos consideráveis em tecnologia e formação, representando um desafio particularmente significativo para as pequenas e médias empresas [8], [9].



Figura 4 – Indústria 4.0 [10]

2.1.2 – Desafios e riscos associados

Os desafios inerentes à transição para a Indústria 4.0 são variados e exigem uma atenção redobrada por parte das organizações. Um dos obstáculos mais significativos é a segurança dos sistemas contra ciberataques, que se tornaram cada vez mais frequentes e sofisticados, pondo em risco a integridade dos dados e a continuidade operacional. Um desafio adicional reside na necessidade de aumentar as competências digitais dos operadores, preparando-os para se adaptarem às novas realidades do ambiente de trabalho, como podemos observar na Figura 5. Estas não

implicam apenas o domínio de novas tecnologias, mas também a adaptação a modelos organizacionais e colaborativos [11], [12].



Figura 5 – Robótica nos dias de hoje [13]

2.1.3 – Planos futuros

A Indústria 4.0 está na vanguarda de uma transformação que promete mudar os processos industriais, impulsionando a eficiência e a inovação. Esta revolução requer uma adaptação consciente, investimentos em formação de operadores e uma reflexão profunda sobre as suas implicações éticas e sociais, sendo fundamental uma colaboração entre empresas, governos, instituições educativas e a sociedade, de modo a maximizar vantagens e mitigar riscos [14].

Uma das principais estratégias para avançar para a Indústria 4.0 é a melhoria na forma de como os dados são recolhidos e processados, recorrendo à automação para obter informações precisas em tempo real. Este tipo de abordagem não só facilita a tomada de decisões, como também impulsiona a automação de tarefas, promovendo eficiência e o crescimento.

Além disso, a Indústria 4.0 procura automatizar a produção a através da modernização das linhas de produção e de distribuição com dispositivos de processamento de dados e programas informáticos inteligentes. Esta integração proporciona flexibilidade na produção, permitindo a personalização dos produtos de acordo com as necessidades dos clientes ao mesmo tempo que se reduz o armazenamento excessivo. Por fim, a Indústria 4.0 não tem apenas o objetivo de melhorar a eficiência, mas também de promover a sustentabilidade, permitindo que as indústrias encontrem formas de economizar energia e recursos para benefício do meio ambiente [15], [16].

2.2 – Robótica Móvel

Devido ao aumento do consumo e à maior oferta de produtos a nível mundial, as indústrias enfrentam pressões, que as obrigam a estar em constante desenvolvimento e à procura de novas tecnologias para satisfazer as necessidades de produção [16], [17]. Uma das formas de conseguir aumentar a produção industrial é através do uso da robótica móvel para distribuir materiais e as ferramentas necessárias nas fábricas [18]. Para este efeito, são utilizados robôs móveis denominados como veículos guiados automaticamente ou AGVs, representados na Figura 6, ou robôs móveis autónomos em inglês *Autonomous Mobile Robot* (AMR), Figura 7, em várias indústrias. Nas últimas décadas, novas tecnologias melhoraram o conceito de AGV, e o novo conceito, conhecido como AMR, representa o próximo passo na automação fabril, através da condução autónoma de veículos.

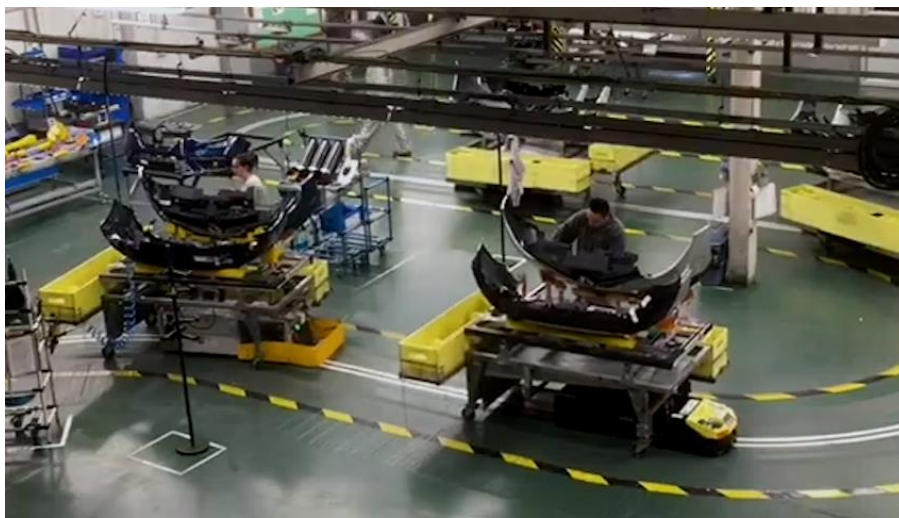


Figura 6 – Os AGVs utilizam uma banda magnética como guia [19]



Figura 7 – Os AMRs não precisam de guia [20]

2.2.1 – Evolução da robótica móvel

O desenvolvimento dos AGVs começou em 1953, com a introdução do primeiro AGV conhecido, pela *Barrett Electronics*, que revolucionou a logística e o transporte de materiais e produtos em ambientes industriais. Inicialmente, o AGV da *Barrett Electronics*, visível na Figura 8, era uma modificação de um trator de reboque, que utilizava uma tecnologia baseada em fios elétricos suspensos no teto da fábrica, permitindo que o AGV seguisse um percurso predefinido. Posteriormente, a tecnologia evoluiu, e a segunda versão deste AGV passou a usar fios colocados no chão, o que simplificou a instalação e a manutenção deste sistema, contribuindo para o seu sucesso em ambientes industriais [21].



Figura 8 – AGVs da Barret Electronics (versão 1 e 2) [22]

Um marco importante na história da robótica móvel foi a criação, em 1975, do primeiro AGV de carga unitária, apresentado na Figura 9. Este modelo era capaz de completar mais tarefas do que qualquer outro AGV desenvolvido até à data, pois estava ligado a um computador.



Figura 9 – Primeiro AGV de carga unitária [23]

Em 1979, os AGVs começaram a navegar com recurso a hardware externo, nomeadamente uma fita magnética instalada no chão da fábrica, conforme é possível

observar na Figura 10, tornando a navegação mais precisa. Esta solução tem, no entanto, a desvantagem de limitar a flexibilidade do layout para mudanças no futuro.



Figura 10 – AGV a utilizar uma banda magnética para a navegação automática [24]

Em 1989, pela primeira vez, deixou de ser necessário utilizar fios ou bandas magnéticas para guiar os robôs móveis, uma vez que estes se guiavam com o auxílio de lasers, conforme é possível observar na Figura 11. Desde então, os AGVs desempenham um papel fundamental no processo de distribuição de materiais em ambientes industriais, sendo frequentemente utilizados em aplicações industriais para mover materiais pelo chão de fábrica ou num armazém.



Figura 11 – Primeiro AGV a utilizar um sistema de lasers para a navegação automática [25]

A última geração de AGVs incorpora tecnologias que lhes conferem a capacidade de comunicação com outros sistemas na mesma fábrica, através da tecnologia *Machine to Machine* (M2M), bem como a integração com sistemas de gestão de armazéns e

gestão de rotas. Isso permite otimizar as rotas e melhorar a eficiência de produção. As rotas podem ser equipadas com sistemas de semáforos, que os AGVs interpretam, parando quando necessário para dar prioridade a outro AGV que necessite de passar pelo mesmo percurso. As últimas gerações de AGVs incorporam também sistemas avançados de segurança, incluindo detecção de obstáculos, prevenção de colisões e sistemas de paragem de emergência, os quais garantem a segurança dos produtos e operadores.

Recentemente, os AMRs têm tido um papel de destaque, nomeadamente os que incorporam tecnologias SLAM que possibilitam uma navegação autónoma precisa e flexível, com base na informação de câmaras de visão, *Light Detection And Ranging* (LiDAR) e outros sensores. Estes AMRs são capazes de mapear e entender o seu ambiente em tempo real, adaptando-se a mudanças de layout e obstáculos inesperados. Na Figura 12, é possível observar as diferenças de comportamento na presença de um obstáculo, entres os dois tipos de robôs móveis mencionados.

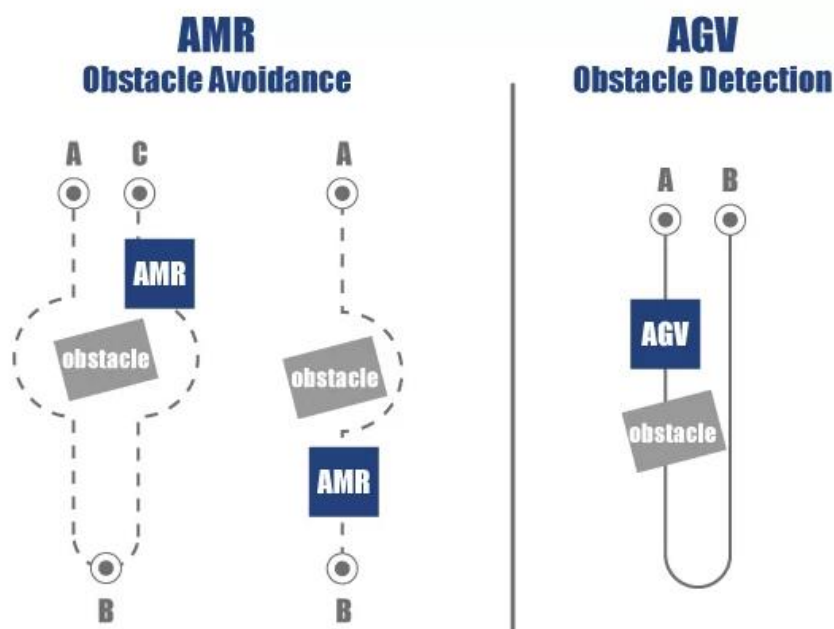


Figura 12 – Diferença entre a deteção de um obstáculo entre A e B [26]

Apesar destas inovações, algumas fábricas ainda utilizam os sistemas de navegação de AGVs por banda magnética, pois estes são mais fiáveis e fáceis de implementar. A Tesla, por exemplo, chama a estes AGVs, apresentados na Figura 13, de “*dumb robots*” e utiliza este sistema de navegação principalmente porque requer menos investimento inicial e qualquer operador com poucas horas de formação é capaz de colocar o sistema a funcionar em caso de avaria.



Figura 13 – AGVs utilizados numa das fábricas da TESLA [27]

No entanto, indústrias como a Amazon dependem de AGVs e AMRs mais sofisticados, apresentados na Figura 14, equipados com tecnologia avançada, para otimizar as suas operações logísticas.



Figura 14 – Robôs móveis utilizados num dos armazéns da Amazon [28]

O futuro dos AGVs e AMRs promete avanços contínuos em tecnologias de navegação autónoma, interligação entre diferentes robôs móveis e conexão com sistemas de gestão de frotas, tornando os AGVs cada vez mais versáteis e eficientes. Novas tecnologias, como a inteligência artificial e o *machine learning* estão a ser utilizadas para melhorar o desempenho dos AGVs, mantendo-os atualizados para suportar todas as evoluções na indústria.

2.3 – Automated Guided Vehicles - AGVs

Os AGVs, visualizados na Figura 15, são veículos automáticos amplamente utilizados na indústria para realizar o transporte independente de materiais em ambiente predefinido sem a necessidade de intervenção humana [29]. Estes veículos estão equipados com diversos tipos de sensores que lhes permitem navegar pela instalação e evitar obstáculos de forma autónoma. Em geral, a seleção do tipo de AGV é feita de acordo com as necessidades específicas de cada aplicação, tendo em consideração diferentes fatores como o tipo e tamanho das cargas a serem transportadas, o layout da instalação e os requisitos operacionais [29], [30].

Entre os tipos mais comuns de AGVs, encontram-se [31]:

- Tipo padrão: caracteriza-se pelas suas funcionalidades básicas de transporte, sendo adequado para aplicações que necessitam de movimentar cargas de um ponto para outro, sem necessidade de operações complexas;
- Tipo reboque: utilizados para transportar materiais ou mercadorias em plataformas, ideais para mover cargas grandes e pesadas, como paletes de mercadorias ou máquinas pesadas;
- Modelos de carga unitária: desenvolvidos para transportar cargas individuais, como paletes e contentores, utilizados em armazéns e centros de distribuição;
- Tipo empilhadora: equipados com mecanismo de elevação para manipular cargas, ideais para operações de produção e armazenamento que requerem manuseamento vertical de materiais.



Figura 15 – Gama de AGVs da ASTI/ABB [32]

2.3.1 – Vantagens e desvantagens

Os AGVs são altamente versáteis e flexíveis, adequando-se a uma variedade de tarefas industriais e logísticas, desde a gestão de armazéns até à otimização de processos logísticos. Uma aplicação inovadora é a abordagem “*Goods to Person*”, na qual os produtos são entregues diretamente aos operadores, demonstrando a eficiência e a versatilidade dos AGVs em responder aos requisitos dos ambientes industriais modernos e em constante evolução.

Na indústria, os AGVs desempenham um papel crucial em aplicações que envolvem a movimentação, manipulação e a carga de materiais. A utilização de AGVs na indústria oferece, entre outras, as seguintes vantagens [33]:

- Redução do esforço físico despendido pelos operadores;
- Deslocamento não tripulado;
- Repetibilidade e produtividade;
- Colaboração com operadores;
- Custos mais controlados e reduzidos;
- Segurança e redução dos danos provocados pelo transporte manual do material de produção;
- Fornecimento de material de acordo com o fluxo da produção, resultando num controlo mais eficiente.

No entanto, os AGVs também têm as suas desvantagens, incluindo algumas limitações quando estes são utilizados em ambientes exteriores, a necessidade de manutenção e de suporte contínuo, bem como a adequação principalmente a tarefas repetitivas.

2.3.2 – Segurança

A segurança dos AGVs é primordial para a otimização de processos em ambientes industriais, garantindo a eficiência operacional e a proteção no local de trabalho. Através da incorporação de tecnologias avançadas, os AGVs conseguem operar de forma segura e eficaz, reduzindo os riscos associados à sua movimentação em espaços partilhados com trabalhadores e outras máquinas.

Para cumprir as normas de segurança, os AGVs são projetados de acordo com as normas europeias para veículos autónomos, incorporando uma série de dispositivos e sistemas de segurança, entre os quais se destacam [34]:

- Detecção de colisões: os AGVs estão equipados com sensores que lhes permitem identificar objetos na sua trajetória, como é possível observar na Figura 16. Em caso de risco de colisão, é possível que os AGVs ajustem a velocidade ou parem completamente para prevenir colisões.

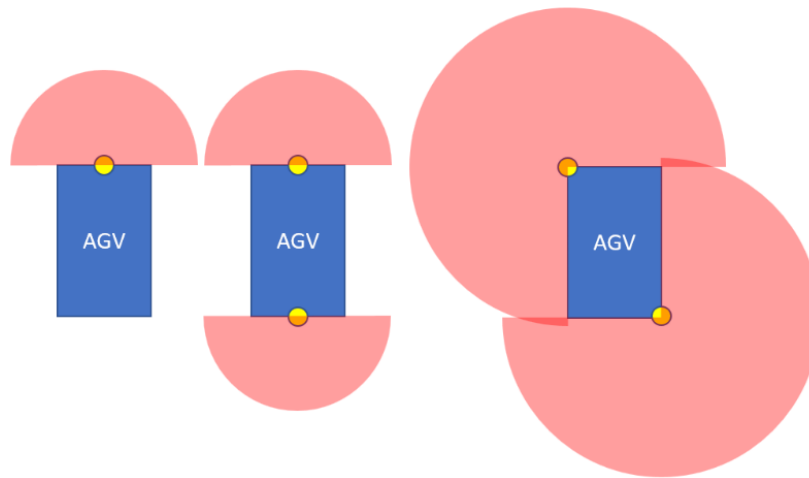


Figura 16 – Ângulos de deteção

- Sinalização áudio/visual: para alertar sobre a sua presença e estado operacional, os AGVs dispõem de sistemas de sinalização que incluem luzes intermitentes, visíveis nas Figuras 17 a 19, bem como alertas sonoros.



Figura 17 – Luz vermelha, indicando anomalia ou modo de emergência



Figura 18 – Luz amarela, indicando modo autónomo



Figura 19 – Luz azul, indicando modo manual

- Controlo manual (Figura 20) e botões de emergência (Figura 21): Estas funcionalidades permitem intervenções manuais rápidas, proporcionando uma paragem imediata em situações de emergência.



Figura 20 – Comando para modo manual



Figura 21 – Botoneira

Para garantir a eficácia destas tecnologias, os AGVs devem ser equipados com:

- Laser scanner e LiDAR: sensores de segurança baseados em tecnologia laser que permitem monitorizar a presença de objetos ou operadores humanos na área de alcance do AGV, criando uma zona de segurança à sua volta, como demonstrado na Figura 22. O *LiDAR* oferece, geralmente, maior alcance e precisão em comparação com o laser scanner como é possível observar na Figura 23, embora este último possa ser mais económico.



Figura 22 – Zona de segurança [35]



Figura 23 – Laser de segurança SICK [36]

- Câmaras: as câmaras, como as apresentadas na Figura 24 e Figura 25, são uma ferramenta importante de segurança nos AGVs, pois são utilizadas para identificar objetos e pessoas. No entanto, em ambientes com pouca luz, a sua precisão pode ser inferior.



Figura 24 – Câmera Alvium U [37]



Figura 25 – Câmera SICK SafeVisionary 2 [38]

- Sensores ultrassônicos: mais econômicos que o *LiDAR* e o scanner laser, os sensores ultrassônicos têm um alcance e uma precisão inferiores. Ainda assim, desempenham um papel relevante na deteção de obstáculos, como podemos observar na Figura 26 e Figura 27.

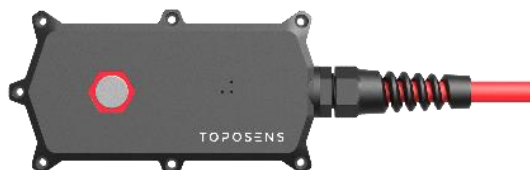


Figura 26 – Sensor ultrassónico [39]

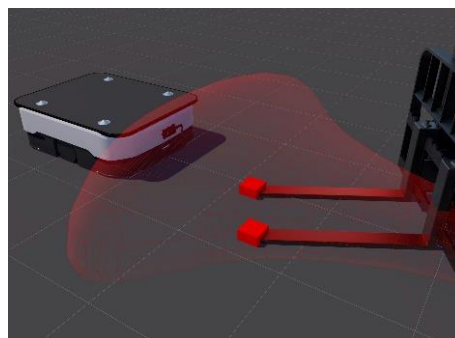


Figura 27 – Zona de ultrassons [40]

- Para-choques (também designados por *bumpers*): são equipamentos de segurança instalados nos AGVs. Conforme demonstrado na Figura 28 e Figura 29, os para-choques interrompem o movimento do robô ao detetar qualquer contacto físico, evitando assim acidentes.



Figura 28 – AGV com para-choques de fábrica



Figura 29 – AGV com para-choques adicionado posteriormente [41]

A integração destas tecnologias permite que os AGVs naveguem, de forma autónoma e segura, cumprindo os padrões de segurança estabelecidos e contribuindo para um ambiente de trabalho mais seguro e eficiente.

Os padrões de segurança são compostos pelas normas internacionais e europeias para robôs móveis, as principais normas incluem:

Normas internacionais:

- ISO 3691-4:2023 – Esta norma define os requisitos de segurança para o projeto, construção, operação e manutenção dos AGVs.
- ISO 13849-1:2023 – Esta norma estabelece os princípios gerais para o projeto seguro dos AGVs.
- IEC 62061:2021 – Esta norma define os requisitos de segurança para o equipamento elétrico dos AGVs.

Normas europeias:

- EN 1175:2020 – Requisitos de segurança para plataformas de elevação de pessoas em AGVs.
- 2006/42/CE – Esta diretiva estabelece os requisitos essenciais de segurança para AGVs.
- EN ISO 3691-4:2020 – Esta norma é a versão europeia da norma ISO 3691-4:2023.

2.3.3 – Movimentação

A movimentação nos AGVs é um aspeto crucial que define a sua eficiência e aplicabilidade em diversos ambientes industriais. A escolha do tipo de movimentação depende de vários fatores como o ambiente de trabalho, o tipo de carga a transportar e as necessidades da operação em questão.

Os sistemas de rodas utilizados nos AGVs variam significativamente, cada um adaptado a necessidades específicas. Os tipos de rodas mais utilizados em AGVs podem ser analisados na Tabela 1 [42].

Tabela 1 – Tipos de Rodas em AGVs

<p>Roda Fixa</p>	 <p>Figura 30 – Roda Fixa [43]</p>	<p>Oferece uma direção constante.</p>
<p>Roda Livre</p>	 <p>Figura 31 – Roda Livre [44]</p>	<p>Permite que o AGV se ajuste às variações da trajetória sem alterar a direção na qual o motor está a aplicar força.</p>
<p>Roda Motriz</p>	 <p>Figura 32 – Roda Motriz [45]</p>	<p>Responsável pela tração, transforma energia elétrica em movimento.</p>
<p>Roda Direção/Motriz</p>	 <p>Figura 33 – Roda Direção/Motriz [46]</p>	<p>Combina a capacidade de direção com a tração.</p>
<p>Roda Omnidirecional</p>	 <p>Figura 34 – Roda Omnidirecional [47]</p>	<p>Possibilita a movimentação em qualquer direção.</p>

Dada a diversidade de aplicações possíveis, os AGVs podem ser classificados em diferentes tipos com base no seu sistema de movimentação, como os três tipos apresentados na Figura 35, Figura 36 e Figura 37.

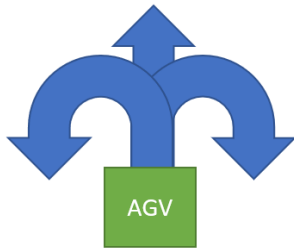


Figura 35 – Unidirecional



Figura 36 – Bidirecional

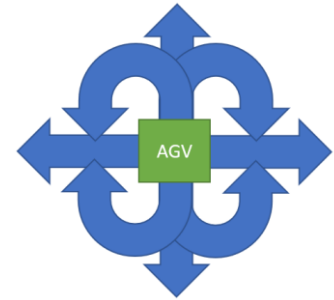


Figura 37 – Omnidirecional

2.3.4 – Rotas

Existem diferentes tipos de rotas que um AGV pode percorrer. Estas rotas são definidas consoante com base nas operações a realizar, nas condições das instalações e na sua localização. A escolha do tipo de rota influencia a escolha do AGV mais adequado para desempenhar tarefas específicas.

Os tipos de rotas habitualmente utilizados são os seguintes [48]:

- Rota em linha – apresentada na Figura 38, este tipo representa a forma mais elementar de uma rota, permitindo ao AGV deslocar-se para a frente e para trás. Esta configuração limita a presença de apenas um AGV por linha.



Figura 38 – Rota com linha única

- Rota em *loop* – esta configuração permite a utilização de vários AGVs em simultâneo que circulem na mesma direção, conforme visualizado na Figura 39.



Figura 39 – Rota em loop

- Rota com vários *loops* – composto por vários *loops* interligados, permitindo assim criar bifurcações e responder a sistemas com várias estações, ou seja, é uma configuração mais complexa, logo, exige controlo de tráfego mais sofisticado, mas oferece maior flexibilidade e otimização de rotas, como é possível visualizar na Figura 40.

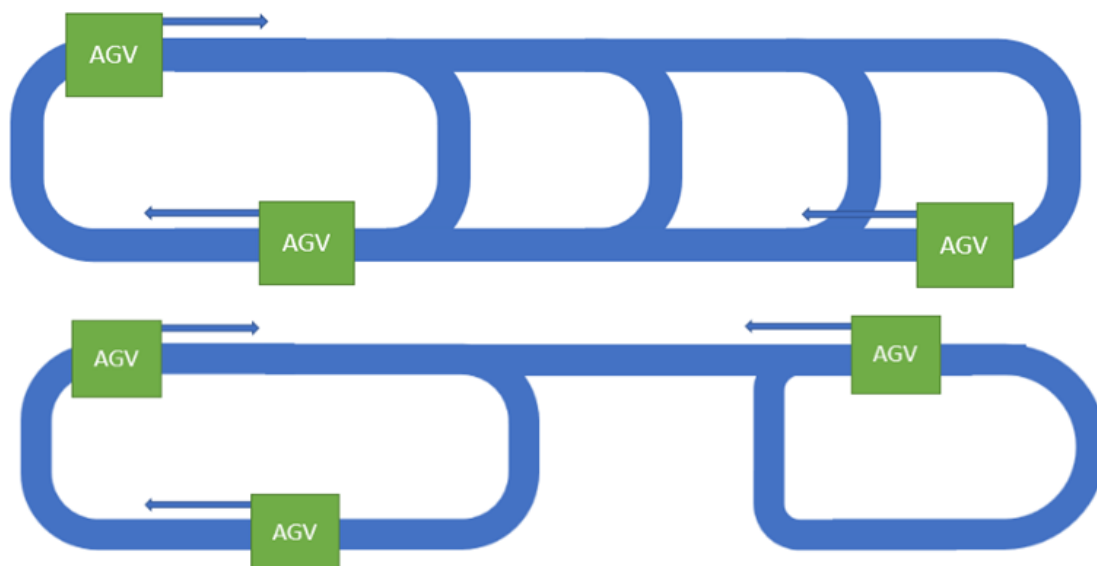


Figura 40 – Rotas com vários loops

A gestão do tráfego torna-se relativamente simples quando a rota é ocupada por um único AGV. No entanto, com a presença de múltiplos AGVs em circulação, é crucial evitar interferências ou colisões. Geralmente, isto é alcançado através da implementação de zonas de exclusividade para prevenir colisões e da criação de ramais estrategicamente posicionados no layout para resolver potenciais conflitos, garantindo assim a fluidez e segurança do sistema.



Figura 41 – Traffic BOX

Também é muito comum utilizar um equipamento como o representado na Figura 41 para validar ordens de movimento, como, por exemplo, quando um reboque acaba de ser carregado ou quando um AGV com prioridade superior precisa de passar em primeiro lugar.

2.3.5 – Carregamento dos AGVs

No contexto dos sistemas de alimentação para os AGVs, é possível identificar uma variedade de tecnologias e métodos de carregamento de baterias. Destacam-se quadro métodos principais:

- Troca manual: este método implica a remoção manual da bateria do AGV para ser recarregada separadamente, ou então a substituição da bateria descarregada por outra previamente carregada.
- Carregamento manual: neste caso, o AGV é ligado diretamente a uma tomada durante períodos de inatividade (fora de produção).
- Carregamento automático: neste método, o AGV regressa automaticamente à estação de carregamento quando a bateria está fraca, iniciando-se o processo de carregamento sem a necessidade de intervenção manual.
- Carregamento sem fios: utilizando uma tecnologia de carga indutiva, este método permite carregar o AGV simplesmente estacionando-o sobre uma superfície equipada com um carregador sem fios.

Cada um destes métodos apresenta vantagens e desvantagens principalmente em termos de eficiência e custo. A escolha do sistema mais adequado depende das necessidades específicas de cada aplicação [49], [50].

2.3.6 – Navegação

Os AGVs utilizam diversos sistemas de navegação para se movimentarem de forma precisa e eficiente no ambiente em que operam. Estes sistemas podem ser distintos, mas os mais comuns, apresentados na Figura 42 são a navegação por banda magnética, a navegação por laser, a navegação por pontos magnéticos e a navegação por SLAM [51].

- A navegação por banda magnética é semelhante à navegação indutiva. Este método utiliza fitas magnéticas coladas no chão da unidade fabril para criar uma rota para o AGV seguir. Existem sensores magnéticos no AGV que detetam a posição da fita e orientam o veículo sobre a mesma. Este método permite uma reconfiguração mais fácil das rotas, reposicionando apenas a banda magnética.
- Na navegação por laser, são utilizados lasers para mapear o ambiente e determinar a posição do AGV em relação a alvos com refletores estrategicamente posicionados. Os lasers fazem um varrimento à área em redor do veículo guiado, a fim de medir todas as distâncias e ângulos do alvo. Embora permita obter um sistema com alta precisão, a navegação por laser pode necessitar de sistemas auxiliares para validação, devido às limitações em ambientes industriais mais complexos.
- Na navegação por pontos magnéticos e fusão de sensores, os AGVs seguem pequenos pontos magnéticos embutidos no chão, e é utilizada fusão de sensores como acelerómetros e giroscópios para calibrar a posição do AGV. Este método proporciona um posicionamento preciso, mas a instalação é mais complexa e invasiva, reduzindo a flexibilidade.
- Na navegação por SLAM são utilizados algoritmos avançados para construir um mapa virtual em tempo real do ambiente em que o AGV se encontra. Este tipo de navegação é altamente flexível e eficiente em vários ambientes, mas os

AGVs que possuem esta funcionalidade apresentam um custo bastante mais elevado.

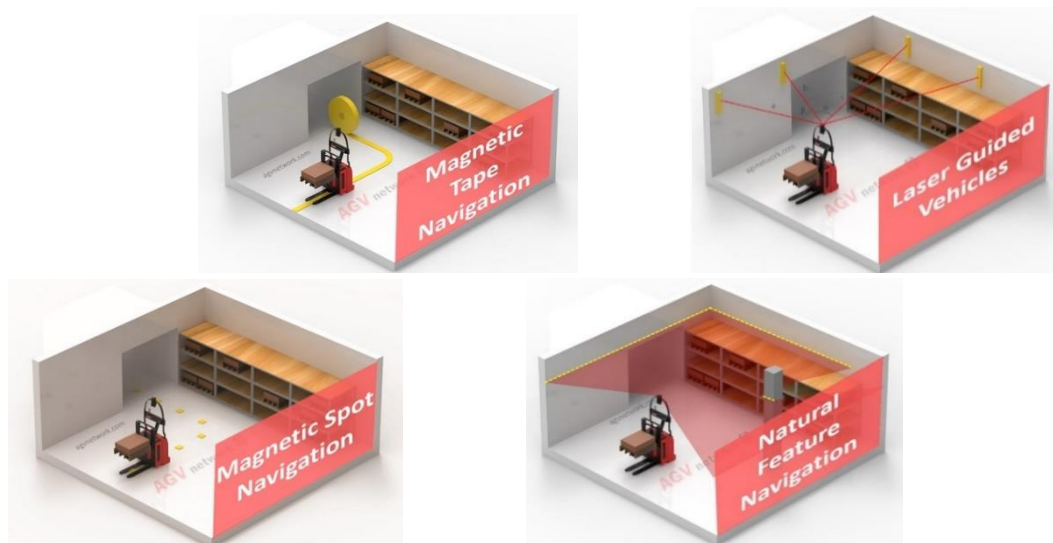


Figura 42 – Principais tipos de navegação em AGVs [52]

2.3.7 – Localização

Cada AGV pode utilizar diferentes sistemas de localização, os quais, apresentam características distintas. Estes sistemas podem incluir tecnologias como tags *Radio Frequency Identification* (RFID), visual, laser e por GPS.

- A localização por meio de tags RFID é uma tecnologia que utiliza tags RFID para determinar a posição do AGV. Estas tags são colocadas em pontos estratégicos no chão da fábrica e emitem sinais de radiofrequência que são recebidos pelo AGV, permitindo que este determine a sua posição exata. Esta é uma abordagem bastante precisa e confiável.
- A localização visual pode utilizar câmaras e algoritmos de visão para localizar o AGV, por exemplo, ao ler um código QR colocado num ponto estratégico seja no chão ou nas paredes. O AGV utiliza uma câmara para detetar a presença destes códigos e assim determinar a sua posição no ambiente. Esta técnica é flexível e pode ser utilizada em diversos ambientes, mas a precisão deste tipo de localização depende da qualidade das câmaras e dos algoritmos de processamento de imagem.
- A localização por laser é caracterizada pelo uso de um scanner laser que, ao longo do trajeto, efetua um varrimento à procura de alvos refletivos. Após obter e processar as distâncias e ângulos relativamente aos alvos, o AGV consegue triangular a sua posição. Esta tecnologia pode necessitar, contudo, de um sistema auxiliar para uma dupla verificação.
- Também é utilizada a localização por GPS, um recurso complementar a outros sistemas de navegação e uma alternativa viável e flexível em contextos de rotas dinâmicas. No entanto, no interior de edifícios, o recetor GPS tende a

apresentar constrangimentos de funcionamento decorrentes do reduzido número de satélites visíveis ou mesmo perda de sinal rádio.

- A localização Inercial utiliza a fusão de sensores, como acelerómetros e giroscópios, para rastrear o movimento do AGV. Embora não seja muito preciso, devido ao acumular de erros ao longo do tempo de funcionamento, este tipo de localização é utilizado com alguma frequência para fornecer dados adicionais sobre o movimento dos AGVs.

2.3.8 – Monitorização

A monitorização dos AGVs é crucial para garantir a sua eficácia e segurança. Uma das tecnologias mais comuns para este efeito é o uso de tags RFID, que consistem em pequenos dispositivos capazes de armazenar informações e de serem lidos sem a necessidade de contacto físico. Os AGVs podem ser equipados com leitores RFID que permitem a deteção e identificação de tags colocadas no chão indicando o percurso que o AGV deve seguir. As tags RFID podem também ser utilizadas para identificar áreas de trabalho específicas, como estações de carregamento, áreas de carga ou descarga, permitindo que o AGV as identifique e que se posicione corretamente para realizar as tarefas designadas.

A monitorização através do uso de tags RFID possibilita também a obtenção de informações em tempo real sobre a localização dos AGVs. No entanto, os AGVs estão sujeitos a perturbações que podem impedir o seu funcionamento adequado, como dificuldades de leitura das tags RFID, que podem comprometer a capacidade do AGV de seguir uma rota predefinida, levando a desvios e possíveis perdas de referência.

Outra tecnologia importante na monitorização de AGVs é a comunicação em tempo real através de redes sem fios. Esta comunicação permite que os AGVs enviem dados de estado e recebam comandos instantaneamente, melhorando a coordenação e a resposta a eventos imprevistos. Alguns sistemas avançados de monitorização podem incluir a análise de dados de sensores nos postos de paragem ou nos próprios AGVs, aumentando a confiabilidade e a eficiência do sistema. No entanto, quando ocorrem perturbações, que derivam da necessidade de manutenção (por exemplo, recarregar as baterias) ou surjam obstáculos no percurso, pode ser necessária a intervenção manual de um operador. Para que isso ocorra de forma eficiente, é essencial que o operador seja informado em tempo real sobre a posição exata do AGV e qual o tipo de problema ocorrido, evitando atrasos e paragens na produção.

Além disso, o uso de câmaras e lasers instalados na unidade fabril, juntamente com algoritmos de visão computacional, pode complementar a monitorização dos AGVs, ao detetar, em tempo real, obstáculos no seu percurso e ajustar a sua trajetória se necessário. A combinação de tecnologias como RFID, redes sem fios e visão computacional fornece uma solução robusta para a monitorização e gestão de frotas de AGVs. Um sistema de gestão de frotas de AGVs é fundamental para fornecer, de forma centralizada, todos os parâmetros relevantes de funcionamento, incluindo o

estado da bateria e a localização de cada AGV, bem como para enviar notificações em caso de situações anómalas que exijam intervenção por parte de operadores.

Para superar as dificuldades associadas à identificação de problemas e à localização dos AGVs numa instalação fabril de grande dimensão, devem ser consideradas soluções baseadas na integração de sistemas. Estas soluções devem ser capazes de recolher informação de cada AGV, independentemente do seu modelo, e apresentá-la de forma agregada através de uma interface comum, facilitando a gestão e a manutenção preventiva da frota, e minimizando o impacto negativo no processo de produção [53], [54].

2.4 – HMI

As interfaces Homem-Máquina (HMI) desempenham um papel crucial na interação entre pessoas e sistemas complexos de dados e operações. Desde os primeiros computadores até aos smartphones mais sofisticados e dispositivos de *IoT*, estas interfaces evoluíram significativamente. Estas interfaces facilitam a gestão de dados e operações, substituindo comandos complexos por interações intuitivas como toques, cliques e comandos por voz. Permitem a comunicação e o controlo eficaz sobre máquinas, aplicações informáticas ou sistemas interligados, proporcionando uma visão em tempo real dos dados operacionais, como podemos observar na Figura 43.

Assim como os controlos e o feedback de um automóvel, onde o condutor interage com vários componentes através de uma interface simplificada, as HMI têm como principal objetivo aumentar a eficiência e a monitorização das operações, permitindo aos utilizadores ajustar facilmente as condições de um equipamento ou monitorizar processos industriais, como a atividade de robôs móveis numa fábrica, promovendo uma gestão mais eficaz e ágil de operações [55], [56].

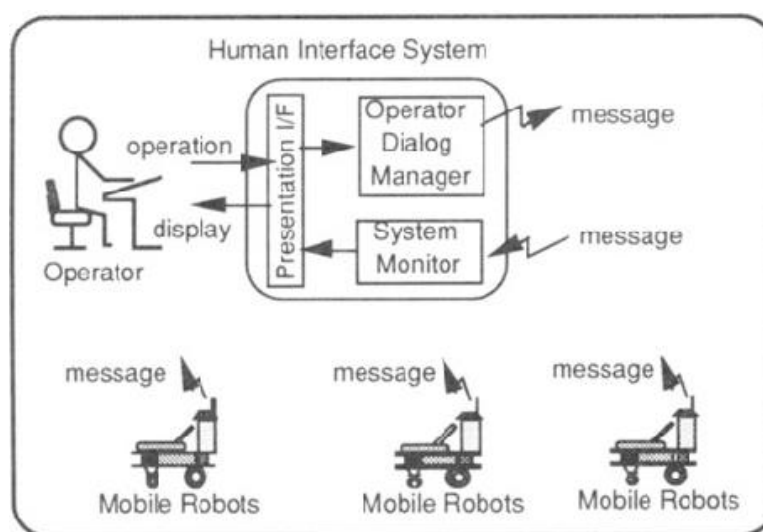


Figura 43 – Lógica das HMIs [57]

2.4.1 – História das HMIs

A história das HMIs é marcada por várias inovações que mudaram a forma como interagimos com diferentes tipos de tecnologia. Tudo começou com o primeiro computador o *Electronic Numerical Integrator and Computer* (ENIAC), apresentado na Figura 44, que utilizava cartões com dados de entrada e devolvia um cartão perfurado como saída [58].

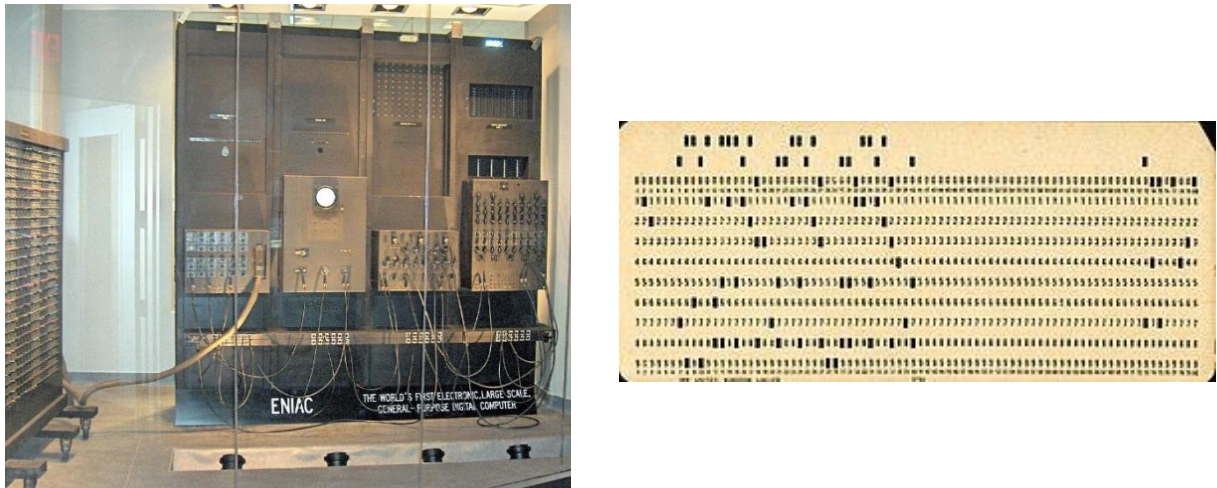


Figura 44 – Primeiro computador ENIAC e os cartões de dados [59]

Os próximos passos ocorreram-se entre os anos de 1960 e 1970 com a introdução de terminais de vídeo (VDTs), que permitiam uma interação mais direta com a utilização de teclados e monitores. Nos anos 80, a Xerox introduziu a interface gráfica de utilizador (GUI) em ambientes industriais com o lançamento do Xerox Alto, Figura 45, o primeiro microcomputador utilizado num HMI industrial, o qual era composto por uma interface gráfica (GUI) e um rato com três botões [60].



Figura 45 – Xerox Alto [61]

Nos anos 90, a incorporação de telas sensíveis ao toque em dispositivos industriais facilitou ainda mais a interação, permitindo o controlo direto e intuitivo de máquinas e sistemas. Esta evolução culminou com a introdução de HMIs em dispositivos móveis, possibilitando a supervisão remota e em tempo real de operações industriais. Nos últimos anos, a pesquisa tem explorado interfaces de comando por voz e gestos em ambientes industriais, ampliando potencialmente as formas de interação e aumentando a flexibilidade e eficiência operacionais, embora a adoção prática ainda esteja numa fase inicial [60].

2.4.2 – Vantagens e desvantagens

As HMIs apresentam várias vantagens que facilitam e otimizam o ambiente de trabalho. Uma destas vantagens é a facilidade de utilização, já que são projetadas com interfaces intuitivas que permitem aos utilizadores interagir de forma simples e eficaz com sistemas complexos, como é possível observar na Figura 46.

As HMIs também possibilitam a monitorização em tempo real dos processos de produção, facilitando a deteção precoce de problemas, como avarias ou a falta de material. Além disso, melhoram a compreensão dos dados, pois organizam a informação de forma mais clara para os operadores, seja através de gráficos ou indicadores visuais.

Além disso, as HMI também ajudam na revisão e otimização contínua dos processos de produção, graças à sua capacidade de monitorização e análise que são capazes de oferecer. Com a otimização e monitorização das tarefas os operadores ganham mais tempo para se dedicarem a outras tarefas com maior prioridade, e para além disso podem trabalhar de uma forma mais segura.

Por outro lado, a utilização de HMIs pode apresentar algumas desvantagens, como o custo de implementação, tanto em termos de equipamentos como de formação de operadores. Além disso, podem também existir problemas de compatibilidade com sistemas mais antigos, o que pode exigir uma análise mais cuidadosa numa fase inicial de planeamento para a integração das HMIs.



Figura 46 – Um operador a utilizar uma HMI [62]

Estes aspetos devem ser cuidadosamente considerados ao avaliar a introdução de tecnologias HMI em ambientes de produção, equilibrando os benefícios de longo prazo com os desafios iniciais da implementação [63].

2.4.3 – Tipos de HMI

As HMI são essenciais em sistemas industriais, podendo assumir várias formas e formatos, como se pode observar na Figura 47, nomeadamente ecrãs encastrados nas máquinas, monitores de várias dimensões e tablets móveis. Independentemente da sua forma, o objetivo de uma HMI é fornecer informação e permitir a interação com várias variáveis de um processo de produção. Os tipos mais comuns de HMI incluem [64], [65]:

- HMI substituta de botões e indicadores: este tipo de HMI substitui botões manuais por um ecrã tátil, permitindo que os trabalhadores interajam facilmente com máquinas e equipamentos. É utilizada para aumentar a produtividade em tarefas simples e repetitivas, reduzindo o esforço físico e a fadiga dos operadores.
- HMI manipuladora de dados: este tipo de HMI recolhe, analisa e processa dados em tempo real, facilitando a compreensão de operações complexas pelos operadores, o que ajuda a melhorar o controlo de qualidade, aumentar a produtividade e a reduzir custos operacionais. É frequentemente utilizada em tarefas complexas que envolvem várias variáveis, como em linhas de produção.
- HMI supervisora: este tipo permite que os operadores monitorizem a fábrica de uma forma macro, obtendo uma visão geral de todo o sistema de operação ou de distribuição, centralizando o controlo. É indispensável em indústrias com uma grande coordenação, como as indústrias de produção em série.

- HMI de visualização: este tipo fornece informação em tempo real a todos os trabalhadores da linha de produção. Os dados são visíveis a vários metros e podem mostrar informações diversas, como indicar que a linha está em pausa devido a uma avaria.



Figura 47 – Diferentes tipos de HMI da marca Siemens [66]

2.4.5 – Aplicações

As HMIs são amplamente utilizadas em diversas indústrias para otimizar a gestão de dados e o controlo de operações, tendo um papel fundamental ao facilitarem a supervisão eficaz e o controlo das operações, garantindo eficiência e segurança. A integração de sistemas de automação permite a interligação entre diferentes sistemas e máquinas, aumentando a flexibilidade e a produtividade, como podemos observar na Figura 48. As HMIs também são essenciais na deteção de falhas, ajudando a identificá-las precocemente minimizando dessa forma tempos de paragem e custos de manutenção. Estas HMIs são capazes de recolher e analisar várias informações para ajudar na tomada de decisões e na melhoria dos processos. Além disso, asseguram a proteção das instalações e dos dados através de sistemas de segurança avançados, como sistemas de controlo de acessos.

Um exemplo de uma aplicação que se encontra em estado de desenvolvimento é a transformação digital que a Chevron está a implementar em conjunto com a Microsoft numa das suas explorações de petróleo e gás. Para monitorizar e controlar operações de perfuração e produção, estão a utilizar sistemas de HMI avançados. Com base nessas informações, os operadores têm a capacidade de visualizar dados em tempo real, dados relativos à pressão dos poços, ao fluxo e a outras variáveis críticas. Esta capacidade permite a deteção precoce de anomalias e possibilita a aplicação imediata de medidas corretivas, reduzindo o risco de acidentes, ao mesmo tempo que aumenta a eficiência das operações [67], [68].



Figura 48 – Tipos de aplicações de HMIs [69]

2.4.6 – Avanços futuros nas HMIs

Os avanços futuros na tecnologia das HMIs prometem transformar ainda mais estas interfaces, tornando-as mais intuitivas, eficientes e imersivas. À medida que as capacidades tecnológicas continuam a evoluir, é possível antecipar várias tendências e inovações que irão ditar o futuro das HMIs. A integração de tecnologias de realidade aumentada (AR) e realidade virtual (VR) nas interfaces, permitirá novas formas de visualização e interação com ambientes de formação mais imersivos, como já está a ser desenvolvido pela BMW desde 2019, com o objetivo de melhorar a eficiência e aumentar o controlo de qualidade, como é possível observar na Figura 49 [70], [71].



Figura 49 – HMI com VR e AR em desenvolvimento pela BMW [72]

As futuras HMI também começarão a incorporar com inteligência artificial e *machine learning*, tornando-se capazes de antecipar necessidades e de se adaptar ao comportamento dos operadores. Além disso, à medida que as HMIs se tornarem mais interligadas e inteligentes, surgirão novos desafios de segurança, exigindo medidas robustas para proteger os sistemas contra ameaças cibernéticas e garantir a integridade dos dados e das operações. A integração das HMIs com a Internet das Coisas (IoT) permitirá uma comunicação contínua e em tempo real entre dispositivos e sistemas, resultando num fluxo de trabalho mais coordenado e eficiente. Estes aspetos refletem as principais tendências esperadas para a evolução das HMIs no futuro, impulsionando a transformação digital em diversos setores e melhorando a interação entre seres humanos e máquinas [71].

2.5 – Sistema flexível de manufatura FMS

O sistema flexível de manufatura, designado por *Flexible Manufacturing System* (FMS), é um sistema de produção que se distingue pela sua capacidade de adaptação e reconfiguração, bem como pelos elevados níveis de automatização incorporados no processo de produção. Este sistema demonstra uma grande flexibilidade, ajustando-se com facilidade a alterações no *design* dos produtos, nas quantidades produzidas, ou na sequência do processo de fabrico, mantendo sempre um nível elevado de eficiência.

Para garantir uma maior produtividade dos sistemas de produção, é essencial efetuar uma gestão eficiente dos dados AGVs, de forma a podemos garantir que o sistema seja capaz de localizar e processar a informação recolhida necessária à tomada de decisões, em tempo útil, conforme apresentado na Figura 50.

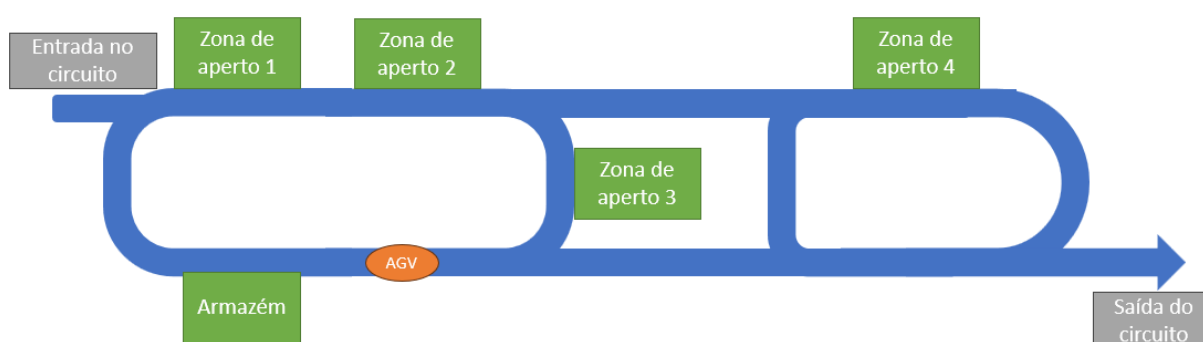


Figura 50 – Exemplo de FMS

2.5.1 – Aparecimento do FMS

O conceito de FMS surgiu na década de 1960, como resposta à necessidade de uma maior eficiência e flexibilidade nas linhas de produção. A primeira implementação prática de um FMS ocorreu na década de 1970, com recurso ao Controlo Numérico Computorizado (CNC), que começou a ser desenvolvido na década de 1950. Desde então, a tecnologia evoluiu significativamente, incorporando avanços na automação, robótica e no controlo computacional, o que permitem uma maior versatilidade e eficiência na produção.

2.5.2 – Vantagens e desvantagens

O FMS constitui um avanço crucial para aumentar os níveis de eficiência e adaptabilidade na indústria. Este sistema destaca-se pela:

- Flexibilidade de produção: O FMS destaca-se pela sua habilidade para fabricar uma vasta gama de produtos sem necessidade de grandes alterações na configuração do sistema, traduzindo-se em economias significativas de tempo e de custos.
- Redução de stock: Graças à sua capacidade para transitar de forma ágil entre diferentes tarefas de produção, os FMS permitem uma diminuição substancial na necessidade de manter grandes quantidades de stock intermédio.
- Elevação da qualidade do produto: Através da automação dos processos, o FMS minimiza os erros humanos, assegurando uma consistência e qualidade superiores.
- Eficiência operacional: A otimização das máquinas e a automação dos processos contribuem para a redução do tempo de paragem e para o aumento do volume de produção.
- Personalização em massa: A adaptabilidade do FMS torna possível satisfazer requisitos específicos de personalização em produtos, sem comprometer a eficiência geral da produção.

No entanto, a implementação de um FMS apresenta desafios significativos:

A aplicação prática de um FMS requer um planeamento extenso, envolvendo a elaboração de projetos e cronogramas detalhados. Além disso, a tecnologia necessita de funcionários qualificados para operar os equipamentos e apresenta um custo inicial mais elevado [73], [74].

2.5.3 – Utilização do FMS em conjunto com a HMI

A utilização combinada de um FMS e da HMI é fundamental para otimizar a produção em fábricas modernas. Na era da automação industrial, a interligação de AGVs em fábricas tornou-se uma prática cada vez mais comum e essencial para aumentar a eficiência da produção. A união de um FMS com uma HMI é crucial neste processo.

A integração eficaz entre um FMS e uma HMI oferece vários benefícios:

- Rastreabilidade e monitorização: a HMI permite uma monitorização detalhada do desempenho de cada AGV, o que ajuda a identificar problemas rapidamente e a tomar as medidas corretivas necessárias.
- Segurança: a HMI também desempenha um papel importante no âmbito da segurança, permitindo que os operadores interrompam ou ajustem as operações em caso de emergência ou situações imprevistas, além de exibir mensagens de erro ou problemas no percurso dos AGVs.

O ponto crucial é implementar uma HMI que trabalhe em conjunto com o FMS para interligar diversos dispositivos, como os AGVs. Isso proporciona maior flexibilidade, permitindo localizar ou consultar informações sobre um AGV específico na fábrica, em tempo real, através do sistema FMS.

DESENVOLVIMENTO

Sistema de supervisão e monitorização de AGVs

Como descrito anteriormente, o objetivo deste projeto é desenvolver uma ferramenta de software para supervisionar e monitorizar a frota de AGVs presente nas instalações da fábrica da Stellantis. Esta solução inclui uma interface HMI composta por várias camadas com diferentes níveis de complexidade, destinada a facilitar a interação com os operadores responsáveis pela gestão dos AGVs. A HMI permite, de forma intuitiva, consultar informação atualizada do sistema de gestão, para tal é essencial uma base de dados bem organizada e estruturada, bem como, uma gestão de acessos para diferentes níveis de utilizadores, garantindo a robustez do sistema.

Um dos principais objetivos do sistema de gestão de frotas é disponibilizar uma aplicação informática que permita visualizar em tempo real os AGVs, mostrando a localização de todos os veículos relativamente à rota selecionada. Esta visualização inclui o estado de cada robô, apresentado por meio de um sistema de cores e informação em tempo real sobre o sentido do seu movimento. Em caso de anomalia ou paragem não programada, o estado do AGV altera-se e é acionado um alarme para informar imediatamente os operadores sobre o problema. Ao selecionar o AGV com anomalia, o operador pode visualizar o robô com mais detalhe, incluindo uma visualização 3D e informações relevantes, como o número identificativo, estado atual e as anomalias existentes.

As informações na aplicação são apresentadas de forma visual e intuitiva, alertando os operadores sobre os diferentes estados dos veículos. Por exemplo, as cores atribuídas a cada AGV correspondem ao código de erro visível no seu visor e indicam o estado do robô: vermelho indica que o veículo está parado devido a uma avaria; azul indica que o veículo está parado, porque não consegue prosseguir na sua rota; e verde representa o normal funcionamento, com o veículo a realizar o seu percurso corretamente. A aplicação permite ainda personalizar este sistema de cores através do menu de configurações, possibilitando a adição de novas cores para monitorizar diferentes tipos de erros.

Além de permitir a monitorização em tempo real, o sistema possibilita também o registo do histórico de movimentações e eventos dos AGVs, oferecendo uma visão detalhada do desempenho da frota ao longo do tempo. Este histórico é essencial para a análise de padrões de falhas, otimização de rotas e para a tomada de decisões informadas sobre a manutenção preventiva dos veículos. Estas funcionalidades, não só visam melhorar a eficiência operacional, como também contribuem para a redução de custos e para a maximização da produtividade.

3.1 – Hardware utilizado

Neste projeto, foram utilizados diversos componentes de hardware, responsáveis pelo processamento dos dados recolhidos nos diferentes AGVs presentes nas instalações da Stellantis e pelo estabelecimento de uma rede local. Os dados serão posteriormente processados com recurso a diversos programas informáticos.

3.1.1 – Servidor

Um servidor como o apresentado na Figura 51 é um computador centralizado que possui um ou vários sistemas integrados, capaz de realizar diversas tarefas, como a recolha de dados, o seu processamento e armazenamento de informações. Neste sistema, o servidor desempenha um papel vital na recolha de dados, em tempo real, de todos os AGVs presentes na unidade fabril. Após a recolha, os dados são processados e enviados para o sistema de supervisão e monitorização desenvolvido, garantindo a eficiência das operações e permitindo uma análise detalhada do desempenho dos AGVs.



Figura 51 – Servidor utilizado no projeto [75]

3.1.2 – Router Wireless

O router, representado na Figura 52, é um equipamento fundamental em redes de comunicação, responsável por encaminhar pacotes de dados entre diferentes redes e assegurar que estes cheguem corretamente ao seu destino. No contexto deste projeto, o router wireless desempenha um papel crítico na estruturação e organização da rede de comunicação, pois foi utilizado para criar uma rede local que interliga todos os dispositivos e sistemas envolvidos no projeto. Isto permite uma comunicação interna eficiente, facilitando a troca de informações entre os pontos de acesso APs, o servidor central e os AGVs. Além disso, o router garante a segurança da rede, implementando protocolos que protegem a rede contra acessos não autorizados e ameaças externas, garantindo a integridade e confidencialidade dos dados transmitidos.



Figura 52 – Router TP-Link Router Wireless utilizado na rede local [76]

3.1.3 – Access points

Os *Access Points* (APs) têm como função principal expandir a cobertura da rede sem fios, permitindo que os dispositivos estabeleçam uma rede local sem a necessidade de cabos. São particularmente importantes em ambientes onde os terminais (AGV) tenham elevada mobilidade e o cenário apresenta obstáculos ou interferências que dificultem a conexão estável.

No âmbito deste projeto, foram utilizados diferentes tipos de APs para garantir uma cobertura eficiente e assegure uma conectividade contínua em toda a área de serviço da unidade fabril. Numa primeira fase foram colocados três APs da *Grandstream*, como é possível observar na Figura 53, em diferentes localizações da unidade fabril, configurados para garantir que todos os AGVs comunicam com o servidor de supervisão central. A escolha destes dispositivos visa garantir que os AGVs mantenham uma comunicação estável e eficiente, essencial para a sua monitorização em tempo real.

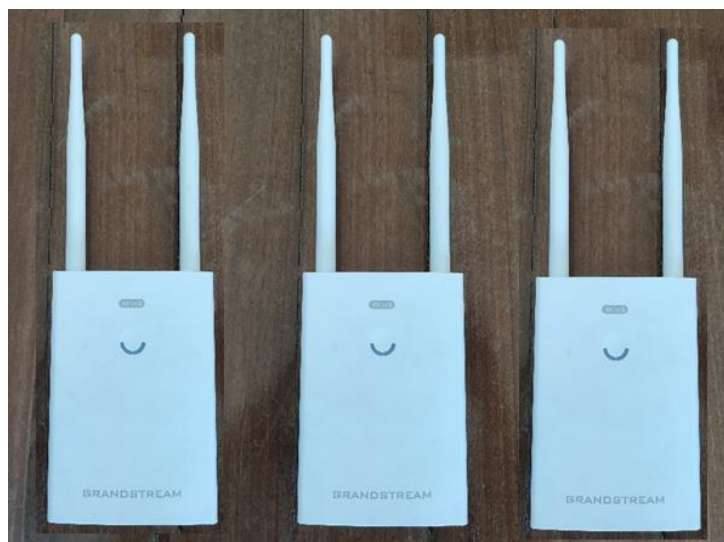


Figura 53 – Access Points da Grandstream

Além dos APs referidos anteriormente, foram também utilizados APs da Acksys, como o representado na Figura 54, que vêm integrados nos próprios AGVs. Estes APs estão ligados aos *dataloggers*, que registam toda a informação relevante sobre o seu funcionamento e desempenho e estão presentes dentro dos AGVs. Os APs da Acksys recolhem estes dados e transmitem-nos para os APs da *Grandstream*, permitindo que as informações sejam centralizadas e processadas no servidor.



Figura 54 – Access Points da Acksys [77]

3.1.4 – Datalogger

O *datalogger*, apresentado na Figura 55, é um dispositivo essencial no processo de recolha e armazenamento de dados provenientes dos AGVs. Este equipamento é responsável por adquirir, registar e armazenar dados de diferentes sensores e dispositivos, permitindo uma análise detalhada posterior.

Cada AGV presente nas instalações da Stellantis possui um *datalogger* ligado via Ethernet a um APs da Acksys, que reencaminha toda a informação registada para ser processada e armazenada no servidor. Esta solução garante que todos os dados recolhidos são transmitidos de forma eficaz e segura, possibilitando uma análise de todas as operações dos AGVs.



Figura 55 – Datalogger existente em cada um dos AGVs [78]

3.1.5 – AGVs

Conforme já descrito neste trabalho, os AGVs, desempenham um papel essencial na automação dos processos industriais, permitindo a movimentação eficiente de materiais nas unidades fabris. Como ilustrado nas figuras 56, 57 e 58, cada AGV está equipado com um *data logger* e um AP, que possibilitam a recolha e transmissão de dados para o servidor central, assegurando a monitorização em tempo real. Esta conectividade é crucial para garantir o funcionamento seguro e eficiente do sistema, reduzindo a necessidade de intervenções manuais e minimizando os tempos de paragem.



Figura 56 – ASTI BidiBot [79]



Figura 57 – ASTI EasyBot [80]



Figura 58 – ASTI TriBot [81]

Os diferentes modelos de AGVs da marca Automatismos y Sistemas de Transporte Interno (ASTI), cujas especificações podem ser consultadas na Tabela 2, variam consoante as necessidades da aplicação. Alguns modelos são projetados para movimentar cargas leves, enquanto outros são robustos para lidar com cargas pesadas ou operar em condições mais adversas. A seleção do modelo de AGV mais adequado depende do tipo e tamanho das cargas a transportar, da complexidade do *layout* da instalação e dos requisitos operacionais.

Embora os AGVs melhorem a segurança e a previsibilidade nas operações logísticas, ao eliminar os riscos associados ao transporte manual, eles apresentam algumas limitações. São mais eficazes para tarefas repetitivas e ambientes internos bem definidos, pois em ambientes exteriores ou em instalações com *layouts* dinâmicos, podem enfrentar desafios, como dificuldades de navegação. Assim, é essencial uma avaliação cuidadosa das condições de operação para maximizar os benefícios da implementação dos AGVs na indústria.

Tabela 2 – Principais características dos AGVs

Descrição	ASTI BidiBot	ASTI TriBot	ASTI EasyBot
Capacidade de carga máxima (kg)	2000	5000	6000
Dimensões (mm)	2104 x 500 x 230	1221 x 695 x 762	1681 x 273 x 600
Movimento	Bidirecional AGV coloca-se por baixo da carga a rebocar	Unidirecional Sistema de reboque tradicional	Unidirecional automático Bidirecional manual AGV coloca-se por baixo da carga a rebocar
Gama de velocidade	0.01 até 1.5 m/s	0.035 até 2 m/s	0.1 até 0.83 m/s
Precisão de posicionamento (mm)	± 10	± 10	-
Movimento manual	Comando de controlo, Wireless	Comando de controlo, Wireless	Comando de controlo, Wireless
Sistemas de segurança	-	-	-
Laser de segurança	2 unidades	1 unidade	1 unidade
PLC de segurança	1 unidade	1 unidade	1 unidade
Botão de emergência	2 unidades	4 unidades	1 unidade [3]
Navegação	Magnético / SLAM / RFID	Magnético / SLAM / RFID	Magnético / RFID
Sistema de Energia	-	-	-
Bateria	Iões de Lítio	Iões de Lítio	Chumbo-ácido
Tipo de carregamento	Carregamento online ou carregador externo	Carregamento online ou carregador externo	Carregamento online ou carregador externo
Comunicação	WiFi, radio, M2M	WiFi, radio, M2M	WiFi, radio
Conectividade	2xHMI, WiFi, USB & Ethernet, ERP, MES, WMS, tablets, PCL interface (OPO)	2 x USB, Ethernet, IoT: ERP, MRP, MES, DDBB interface <i>Industrial connectivity: OPC, wired connection</i>	WiFi, Ethernet, PCL interface (OPO)

3.1.6 – Banda magnética e tags RFID

Para que os AGVs se movimentem de forma automática por toda a fábrica e executem as tarefas designadas, é essencial a utilização de bandas magnéticas e tags RFID, conforme ilustrado na Figura 59. As bandas magnéticas funcionam como guias que definem o percurso que os AGVs devem seguir, garantindo uma navegação precisa ao longo das rotas predefinidas. Este método é simples, fiável e ideal para ambientes industriais onde as rotas dos veículos são fixas ou têm variações mínimas.

As tags RFID são posicionadas em pontos estratégicos ao longo do percurso, fornecendo instruções adicionais aos AGVs, como mudanças de rota, paragens específicas ou execução de ações específicas, como a mudança de direção quando existem múltiplas opções de trajeto. Estas tags são lidas por leitores RFID instalados em todos os AGVs, que posteriormente enviam as informações necessárias para que o AGV reaja de forma correta, permitindo-lhes adaptar o seu comportamento às necessidades operacionais, ajustando a velocidade, alterando a direção ou realizando certas operações em determinados pontos.

A combinação de bandas magnéticas e tags RFID permite uma navegação eficiente e segura dos AGVs, aumentando a flexibilidade e a adaptabilidade do sistema de transporte na fábrica. Este sistema integrado garante que os AGVs cumpram as suas funções de forma otimizada, minimizando erros de navegação e assegurando que as rotas designadas são seguidas com precisão, contribuindo para a eficácia global da operação logística.



Figura 59 – Material utilizado (Banda magnética e Tags RFID) [82]

3.2 – Softwares utilizados

Para alcançar o objetivo desta tese, que é o desenvolvimento de uma aplicação web para a monitorização e supervisão de AGVs, será necessário utilizar uma variedade de softwares e ferramentas tanto para testes como para o desenvolvimento.

3.2.1 – Software dos APs da marca Grandstream

O software dos APs da marca Grandstream, apresentado na Figura 60, é uma página web onde é possível configurar e verificar as conexões aos APs que foram utilizados para a comunicação entre os AGVs e o servidor central.

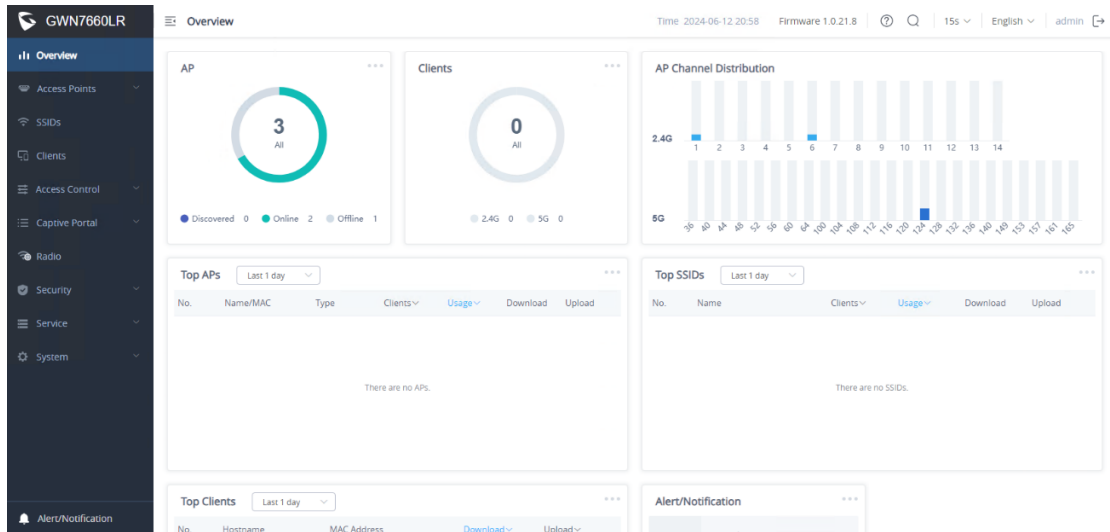


Figura 60 – Página inicial do software dos APs

3.2.2 – AutoCAD

O AutoCAD é um software de design assistido por computador (CAD) desenvolvido pela Autodesk, ilustrado na Figura 61, amplamente utilizado em diversas áreas como arquitetura, engenharia e design gráfico, que permite a criação de desenhos técnicos 2D com precisão e eficiência. Neste projeto, foi utilizado para aceder às rotas pré-definidas pela unidade fabril, bem como para adicionar apontamentos em locais específicos, para posterior processamento.

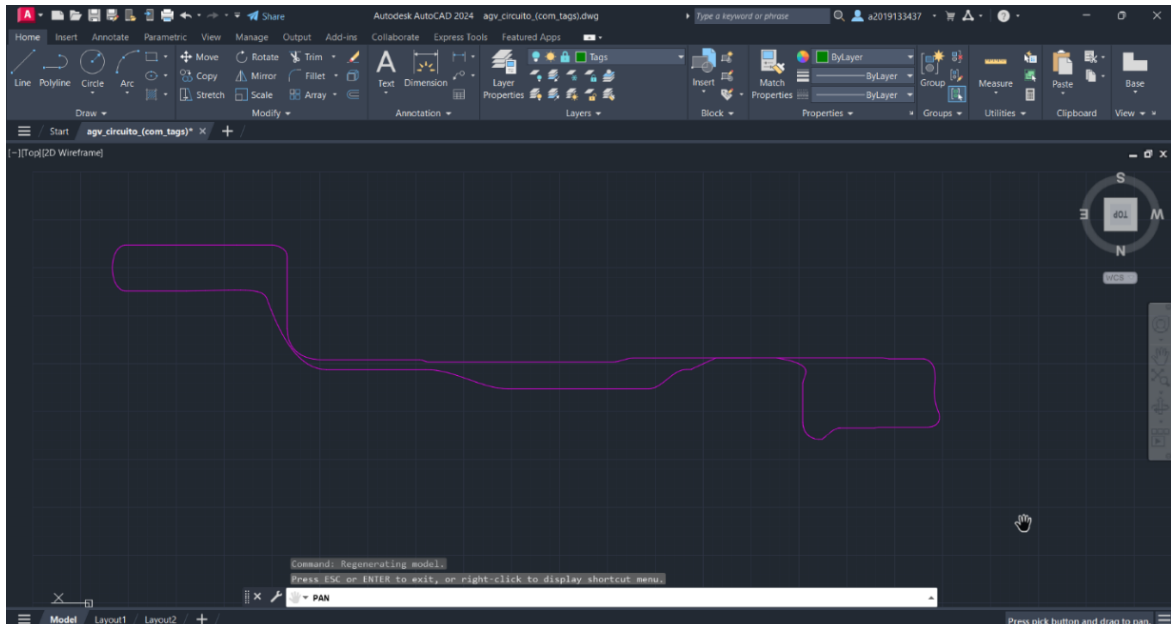


Figura 61 – Aplicação AutoCAD com o exemplo de uma rota

3.2.3 – Aplicação da ASTI

A aplicação dos AGVs da marca ASTI, conforme demonstrado na Figura 62, é uma aplicação web, onde é possível efetuar diversas configurações e recolher registos e logs, que vão ser essenciais na fase inicial do projeto.

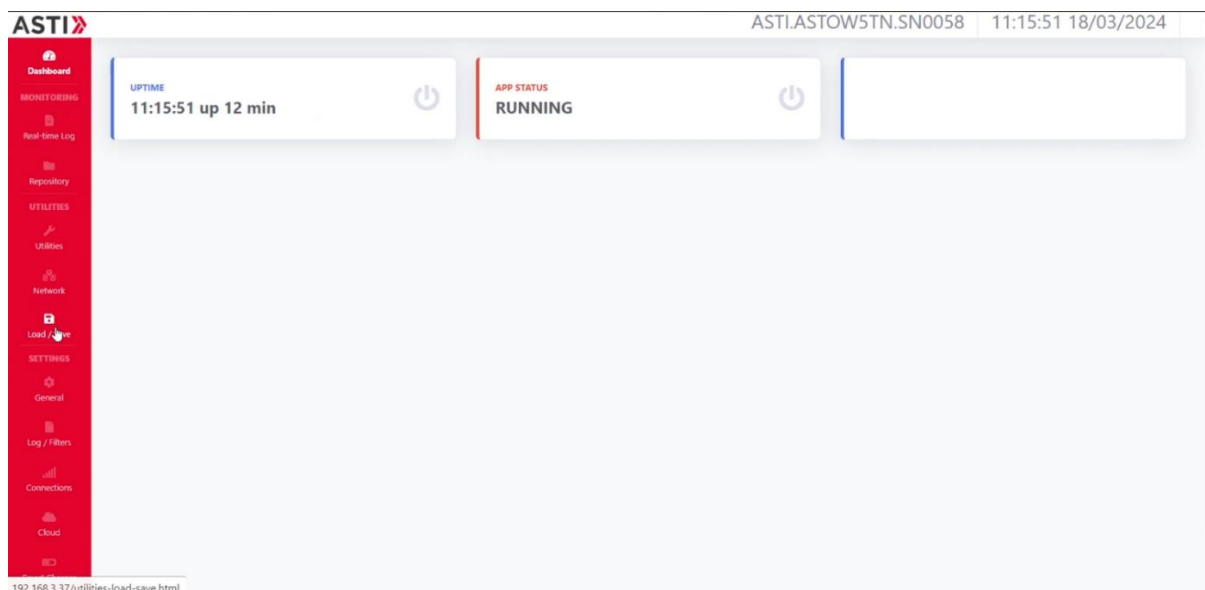


Figura 62 – Aplicação Web da ASTI

3.2.4 – SIGAT

O SIGAT, ilustrado na Figura 63, é uma ferramenta da marca ASTI que permite programar os AGVs. A ligação pode ser feita via cabo Ethernet ou, com as devidas configurações, através de redes Wi-Fi.

AGV	VERSION	ACTIVO	FECHA	RUOTA	PUNTO	SEGMENTO	ZONA A SOLICITADA	ZONA A OCUPADA	ZONA B SOLICITADA	ZONA B OCUPADA	BATERIA	ESTADOTEKTO	INFORMACION	TIPOAGV	ESTADOSOCKET	NIVELBAT	DETALLE
1	1.1	ACTIVO	17/03/2014 10:25:59	1	4	4	0	0	0	0	0 OK	Desconocido	000000000000	1	NDK	242	
2	2.1		1/04/2014 12:19:16	1	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	243	
3	3.1		1/04/2014 12:19:18	1	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
4	4.1		1/04/2014 12:19:19	1	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	252	
5	5.1		1/04/2014 12:19:20	2	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	234	
6	6.1		1/04/2014 12:19:21	2	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	245	
7	7.1		1/04/2014 12:19:22	2	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	244	
8	8.1		1/04/2014 12:19:23	1	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	237	
9	9.1		1/04/2014 12:19:24	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	233	
10	10.1		1/04/2014 12:19:25	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	245	
11	11.1		1/04/2014 12:19:27	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	241	
12	12.1		1/04/2014 12:19:28	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	250	
13	13.1		1/04/2014 12:19:29	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	231	
14	14.1		1/04/2014 12:19:30	3	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	237	
15	15.1		1/04/2014 12:19:31	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	233	
16	16.1		1/04/2014 12:19:32	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	233	
17	17.1		1/04/2014 12:19:34	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	253	
18	18.1		1/04/2014 12:19:35	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
19	19.1		1/04/2014 12:19:36	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	247	
20	20.1		1/04/2014 12:19:37	6	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	237	
21	21.1		1/04/2014 12:19:38	6	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
22	22.1		1/04/2014 12:19:39	6	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	235	
23	23.1		1/04/2014 12:19:40	5	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
24	24.1		1/04/2014 12:19:42	7	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	245	
25	25.1		1/04/2014 12:19:46	7	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	230	
26	26.1		1/04/2014 12:19:44	7	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	253	
27	27.1		1/04/2014 12:30:03	7	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	249	
28	28.1		1/04/2014 12:19:46	8	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
29	29.1		1/04/2014 12:19:46	8	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	237	
30	30.1		1/04/2014 12:19:45	8	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	240	
31	31.1		1/04/2014 12:19:46	8	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	233	
32	32.1		1/04/2014 12:19:46	8	0	0	0	0	0	0	0 OK	Desconocido		1	Unknown	238	

Figura 63 – Aplicação da ASTI (SIGAT)

3.2.5 – SIGAT MultiAGV

O SIGAT MultiAGV, apresentado na Figura 64, é um software da ABB/ASTI que permite a ligação a vários AGVs de diferentes modelos. Além disso, possibilita a recolha de diversas informações dos AGVs em tempo real, que são armazenadas numa base de dados para posterior processamento.

ID/No	AGV	Enabled/Disabled	Communication	IP	Route	Post	Segment	Display Code	State	Zone A-Ashed	Zone A-Occupied	Zone B-Ashed	Zone B-Occupied	Battery Level	Battery Info	Last Message Date
1	AGV 3203	On filter	NOK	Contiene	Equal a	10	10		Disconnected	0	0	0	0	242	0000000	18/05/2024 16:37
2	AGV 3137		NOK	Contiene	Equal a	132	0		Disconnected	0	0	0	0	240	0000000	18/05/2024 14:57
3	AGV 1836		NOK		0	0	0		Available	0	0	0	0	0	0	0
4	AGV 185C		NOK		1	5	1		Disconnected	0	0	0	0	72	0010000	28/06/2024 12:14
5	AGV 2178		NOK		0	0	0		Available	0	0	0	0	0	0	0
6	AGV 3623		NOK		0	0	0		Available	0	0	0	0	100		

Figura 64 – SIGAT MultiAGV

3.2.6 – Software dos APs dos AGVs (Acksys)

O software dos APs dos AGVs (Acksys), apresentado na Figura 65, é uma interface web que permite configurar as conexões aos APs dos AGVs. Estes serão utilizados para comunicar com os APs da rede e enviar posteriormente as informações ao servidor central.

ACKSYS COMMUNICATIONS & SYSTEMS
Wireless just became easier
AirLink series

SETUP TOOLS STATUS

DEVICE INFO NETWORK WIRELESS SERVICES LOGS

DEVICE INFORMATION

FIRMWARE INFORMATION

WaveOs version:	4.10.1.1
Boot loader version:	3.4.2.1
Firmware ID:	E2148.AC.1
SSH access:	enabled (by configuration)

DEVICE INFORMATION

Host name:	ASTL_BB2_0_SNXXXX
Model:	AirLink
Product version:	V1
Motherboard ID:	00001b8e9e95
Product serial number :	19170383

Figura 65 – Software dos APs dos AGVs (Acksys)

3.2.7 – Xampp

O Xampp, cujo logotipo está representado na Figura 66, é um pacote de software que facilita a criação de servidores locais. No âmbito deste projeto, o Xampp foi utilizado para o desenvolvimento inicial, possibilitando a criação de um servidor Apache para hospedar páginas Web locais e um servidor MySQL para criar e gerir bases de dados.



Figura 66 – Logotipo do software Xampp [83]

3.2.8 – PHPMYAdmin

O PHPMYAdmin, conforme mencionado anteriormente e ilustrado na Figura 67, é uma ferramenta de administração de bases de dados MySQL, que oferece uma interface web intuitiva, facilitando a gestão e a manipulação das bases de dados. Permite executar *queries*, gerir tabelas, colunas, relações e outras funcionalidades. No projeto, o PHPMYAdmin foi utilizado para gerir as bases de dados envolvidas, permitindo uma gestão eficiente e eficaz das mesmas e garantindo que todos os dados sejam processados de forma correta.

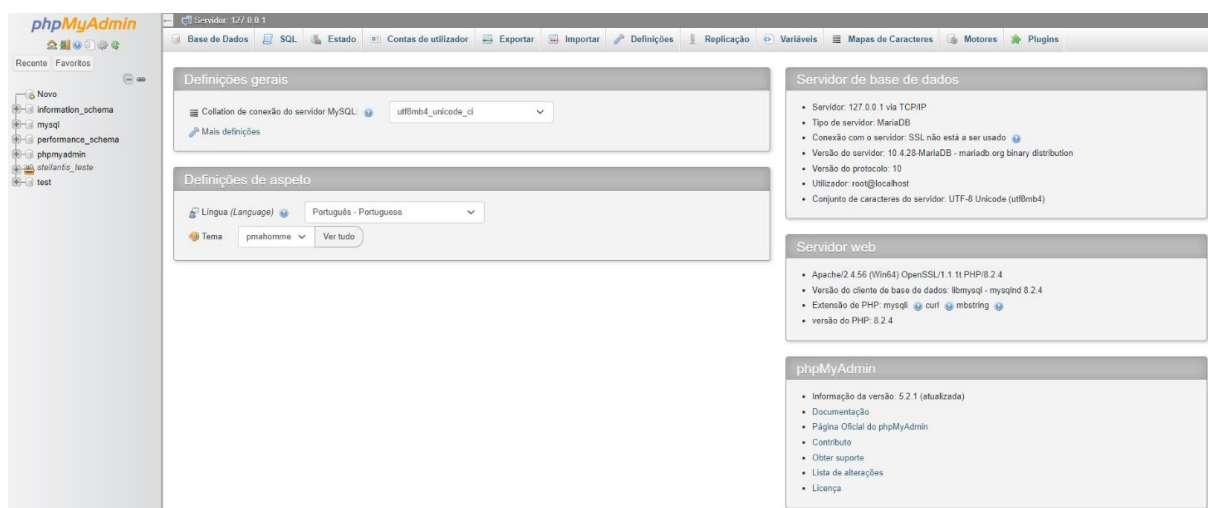


Figura 67 – Página inicial do PHPMYAdmin

3.2.9 – SQL Server Management Studio

O *SQL Server Management Studio* (SSMS), Figura 68, é uma ferramenta que permite consultar e editar bases de dados complexas de forma intuitiva. A sua utilização foi justificada pela complexidade das bases de dados dos AGVs, que dificultam a edição e manipulação direta, tornando o uso do SSMS essencial para essas operações.

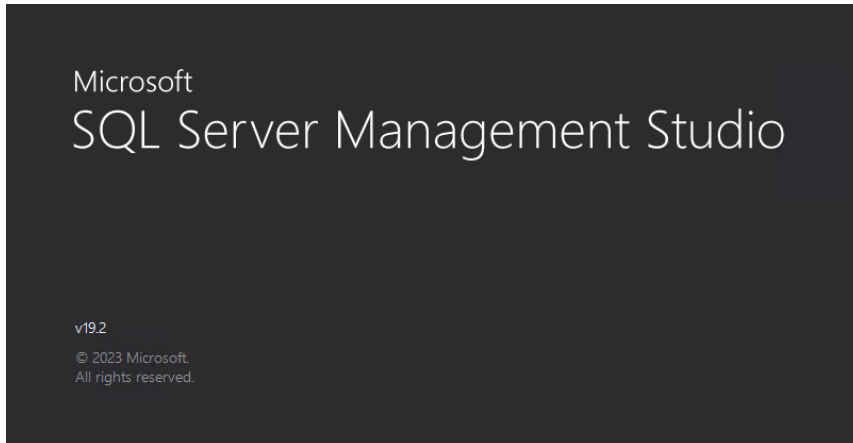


Figura 68 – SQL Server Management Studio

3.2.10 – MySQL Workbench

O MySQL Workbench, Figura 69, é uma ferramenta essencial para criar e estruturar bases de dados, que disponibiliza funcionalidades avançadas para definir chaves, estruturar colunas e estabelecer relações entre tabelas, proporcionando um ambiente visual da base de dados.



Figura 69 – Logotipo do MySQL Workbench [84]

3.2.11 – Linguagem Python

A linguagem Python, cujo logotipo está representado na Figura 70, é uma linguagem de alto nível, interpretada de script e de propósito geral, amplamente utilizada em diversas áreas, como: o desenvolvimento web, a ciência de dados, a inteligência artificial, a automação de scripts, entre outras.

Apresenta várias vantagens em comparação com outras linguagens, nomeadamente uma sintaxe simples e de fácil compreensão, bem como uma grande quantidade de bibliotecas, *frameworks* e repositórios que facilitam muito o processo de programação. Neste projeto, a programação em Python foi utilizada para diversos testes iniciais de lógica devido à sua programação mais rápida e acessível. Após a lógica e os algoritmos funcionarem corretamente em Python, foram convertidos para os tipos de linguagem mais adequados ao efeito e objetivo desejados.



Figura 70 – Logotipo do Python [85]

3.2.12 – Linguagem PHP

O PHP, cujo logotipo está representado na Figura 71, é uma linguagem de script de código aberto, de utilização geral e especialmente adequada para desenvolvimento de páginas web. Trata-se de uma linguagem bastante simples, mas que também dispõe de diversos recursos avançados.

Geralmente, o PHP é utilizado no lado do servidor, que é onde foi utilizado neste projeto para processar e estruturar de corretamente a consulta às bases de dados.



Figura 71 – Logotipo do PHP [86]

3.2.13 – Linguagem JavaScript

O JavaScript (JS), representado na Figura 72, é uma linguagem de alto nível, interpretada, e de propósito geral, amplamente utilizada no desenvolvimento web, como o intuito de adicionar interatividade e dinamismo às várias páginas. Ao contrário do PHP, que é processado no servidor, o JS é executado no cliente.

Neste projeto, as principais utilizações foram a manipulação do documento a partir do *Document Object Model* (DOM) da página web, possibilitando a atualização do

conteúdo de forma dinâmica sem a necessidade de recarregar a página. Além disso, o JavaScript é essencial para implementar tecnologias como o AJAX, que permite comunicação assíncrona com o servidor. Estas tecnologias visam melhorar a interatividade e a experiência do utilizador na aplicação web.



Figura 72 – Logotipo do JavaScript [87]

3.2.14 – Linguagem CSS

O *Cascading Style Sheets* (CSS), cujo logótipo está representado na Figura 73, é uma linguagem de estilo utilizada para descrever a apresentação de um documento escrito em HTML. Ao permitir separar a estrutura do conteúdo da aparência, o CSS facilita a programação e manutenção do design das páginas web.

Com o uso do CSS é possível definir uma série de estilos para os elementos HTML, como cores, fontes, margens e layouts. Neste projeto, o CSS foi utilizado para tornar as páginas mais agradáveis, perceptíveis e consistentes, facilitando a sua utilização.



Figura 73 – Logotipo do CSS [88]

3.2.15 – Linguagem HTML

O *HyperText Markup Language* (HTML), cujo logotipo está representado na Figura 74, é a linguagem padrão para criar documentos web, onde é definida a estrutura e o conteúdo de uma página web. Permite utilizar vários elementos e tags para formatar texto, inserir imagens, importar ficheiros, etc. O HTML é a base de qualquer página web e é essencial para a correto funcionamento dos sites na Internet. Neste projeto, o HTML foi utilizado para estruturar o conteúdo das páginas web e definir os diferentes elementos que compõem a interface do utilizador.



Figura 74 – Logotipo do HTML [89]

Estas ferramentas desempenham um papel fundamental em diferentes fases do projeto com o objetivo de garantir que a aplicação seja robusta, eficiente e capaz de atender às necessidades dos operadores, proporcionando uma solução completa para a monitorização e supervisão dos AGVs em ambiente fabril.

3.3 – Recolha da informação das rotas e tags

No que diz respeito ao desenvolvimento do projeto, este começou com uma análise aprofundada de todas as informações pertinentes para a realização de um estudo prévio. Foram recolhidos diversos dados relacionados com os AGVs, conforme já demonstrado anteriormente. No entanto, o objetivo inicial foi analisar detalhadamente todas as rotas, a posição das tags e as zonas de trabalho, com vista a compreender a base sobre a qual o projeto será desenvolvido.

Para este efeito, procedeu-se à recolha de informações sobre as diversas rotas e a localização das respetivas tags RFID. Além disso, foram recolhidos os mapas virtuais já existentes, como se pode ver na Figura 75, aos quais foi necessário efetuar atualizações e modificações para o desenvolvimento de futuras simulações.

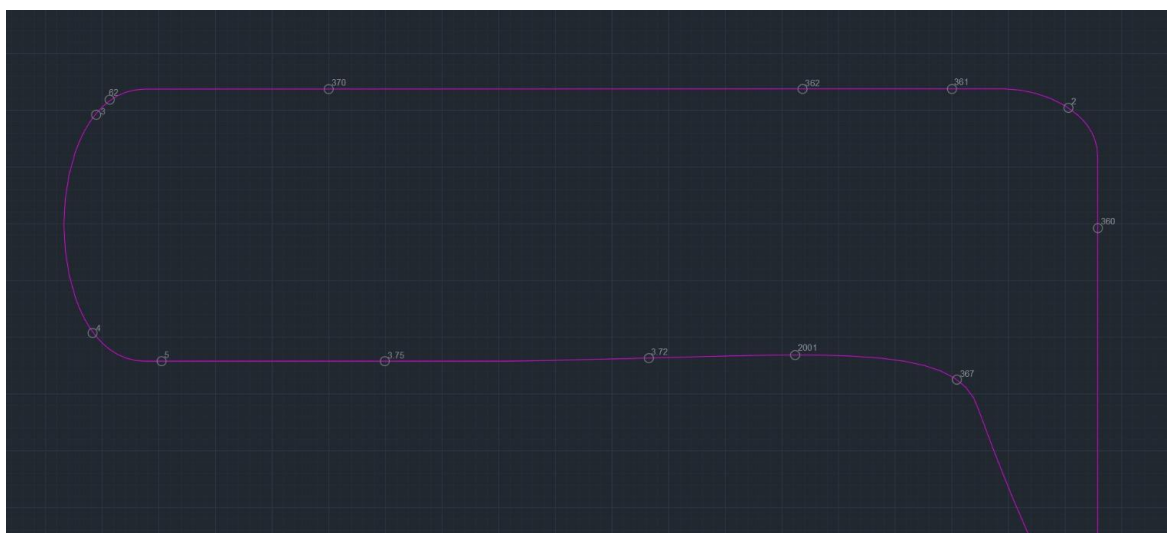


Figura 75 – Excerto de uma rota retirada com as respetivas tags

3.4 – Criação de uma rede com APs

Além da recolha de dados sobre as rotas, foi também realizado um levantamento da quantidade de APs necessários para a criação de uma rede funcional, destinada ao desenvolvimento e aos testes iniciais do sistema de supervisão a ser desenvolvido.

Para este efeito, foi utilizado um router, conforme referido anteriormente, que foi configurado para criar uma rede local, como se pode observar na Figura 76.

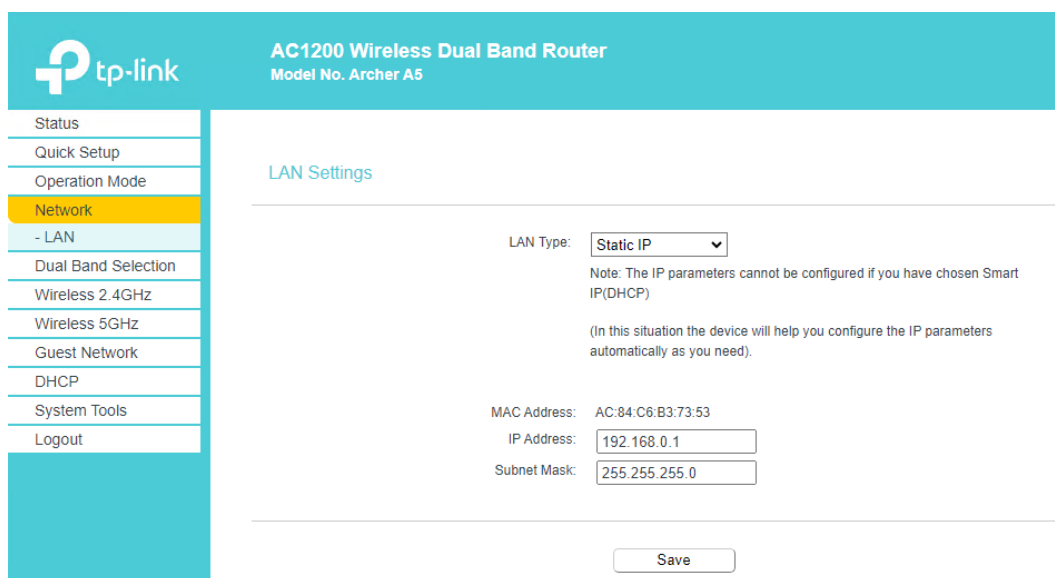


Figura 76 – Configuração do Router utilizado para a criação da rede local

Após a configuração e verificação do funcionamento da rede local, procedeu-se à instalação de vários APs da marca Grandstream, que foram configurados através do respetivo software, conforme ilustrado na Figura 77. Esta configuração permitiu a criação de uma rede *mesh*, que possibilita a mudança dos clientes de um AP com o sinal mais fraco para outro com o sinal mais forte, evitando assim a perda de informação.

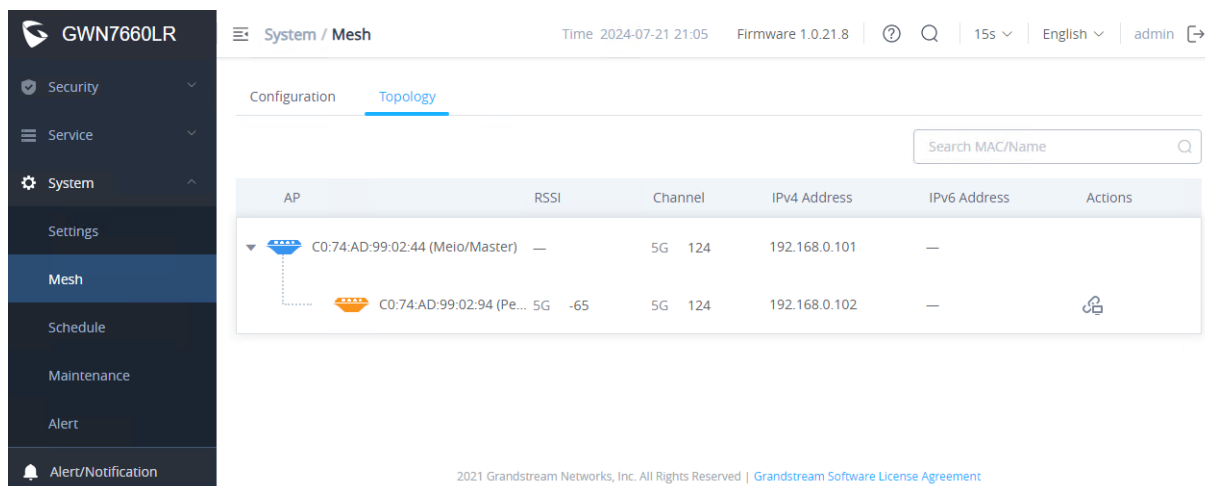


Figura 77 – Configuração dos APs utilizados para a criação de uma rede mesh

Adicionalmente, foram também efetuados testes de alcance aos APs, para determinar o número de equipamentos necessário para cobrir toda a unidade fabril, com vista a garantir uma comunicação eficaz.

Durante estes testes, os APs foram estrategicamente posicionados em diversos locais ao longo da fábrica, de modo a permitir que os dispositivos se conectem ao AP com o nível de sinal de rádio mais elevado.

Este AP, é responsável por encaminhar toda a informação para um ponto de acesso central, permitindo a consulta dos detalhes de cada dispositivo conectado, conforme ilustrado na Figura 78.

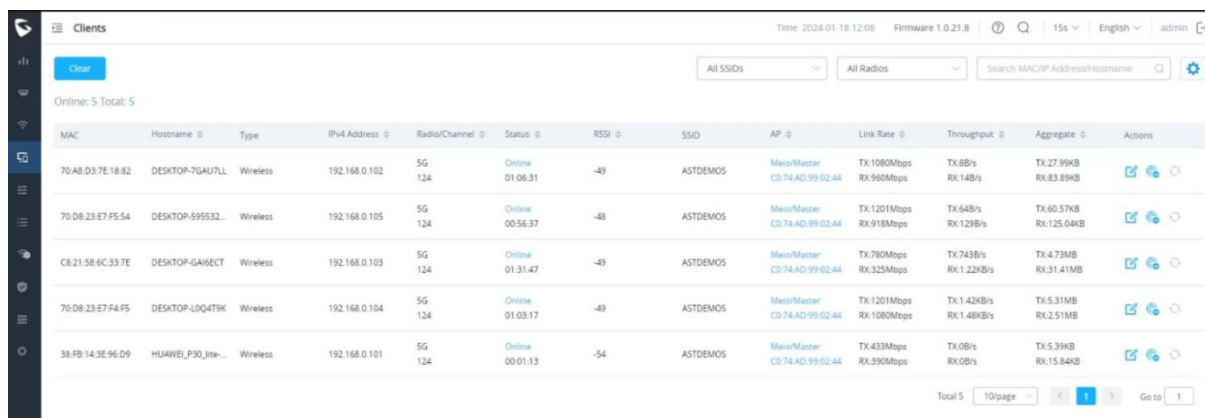


Figura 78 – Detalhes dos equipamentos conectados à rede local

Estes testes permitiram estabelecer uma infraestrutura de rede robusta, capaz de suportar as necessidades iniciais de desenvolvimento e de testes do sistema de supervisão. Assim, o projeto pode avançar para a próxima etapa sem quaisquer preocupações futuras relacionadas com a comunicação na rede.

3.5 – Testes com o AGV

Com a rede de APs em funcionamento, foram realizados alguns testes com os AGVs do tipo apresentado na Figura 79. Estes testes incidiram sobre a análise dos movimentos dos AGVs, nas diversas aplicações em que estes podem ser utilizados e na sua programação.



Figura 79 – AGV com mais abundância na Stellantis [90]

Após conseguir movimentar o AGV e perceber o seu funcionamento básico, foram também criados alguns programas curtos na plataforma de programação SIGAT. Nesta aplicação podemos definir diversas variáveis, conforme ilustrado na Figura 80.

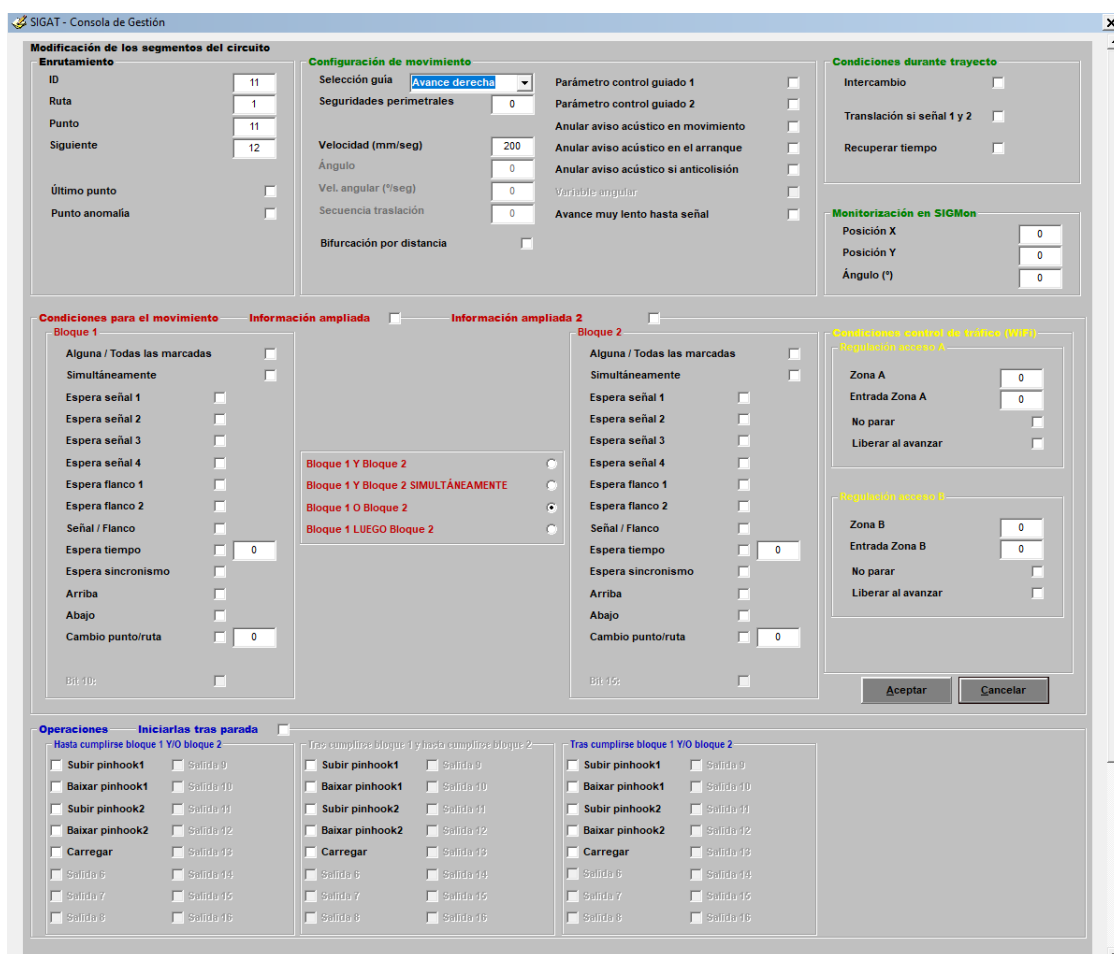


Figura 80 – Várias opções de programação na aplicação SIGAT

Para testar ao máximo as funcionalidades do AGV, foi criada uma rota temporária para testes, conforme é possível observar na Figura 81. O AGV foi programado com tarefas simples, o que permitiu compreender toda a lógica do seu funcionamento e identificar eventuais dados essenciais.

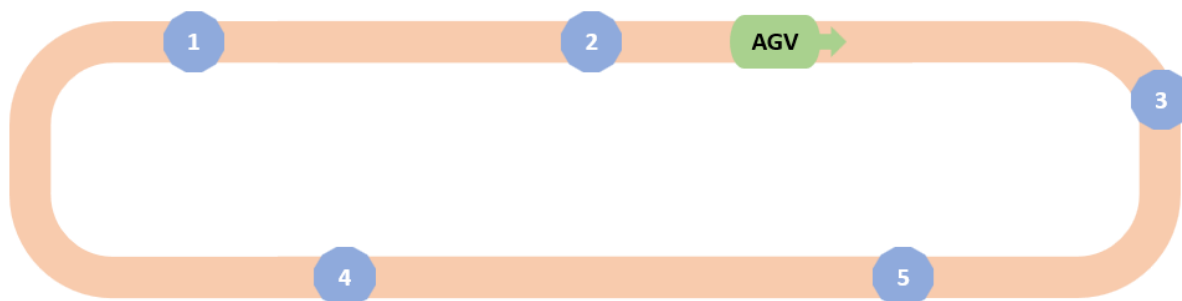


Figura 81 – Exemplo de rota utilizada para testes

3.6 – Recolha de dados dos AGVs

3.6.1 – Versão mais recente dos AGVs

No desenvolvimento da aplicação de supervisão e monitorização dos AGVs, o principal desafio reside na obtenção e gestão dos dados provenientes destes veículos, sendo este o fator mais crítico do processo. Para otimizar o acesso à informação, foram selecionados AGVs equipados com a versão mais recente de *dataloggers*, que disponibilizam um maior volume de dados e funcionalidades de monitorização.

Inicialmente, optou-se por ligar os AGVs diretamente ao computador portátil com cabos *Ethernet*, e por ligar o *datalogger* diretamente ao computador, a fim de reduzir os erros de comunicação. Esta abordagem permitiu evitar a utilização da rede wireless existente, garantindo uma ligação estável e fiável durante as fases iniciais de configuração e testes. Para estabelecer comunicação com o *datalogger*, foi necessário configurar o endereço IP do computador para que este corresponda à gama de endereços IP apropriada, conforme ilustrado na Figura 82.

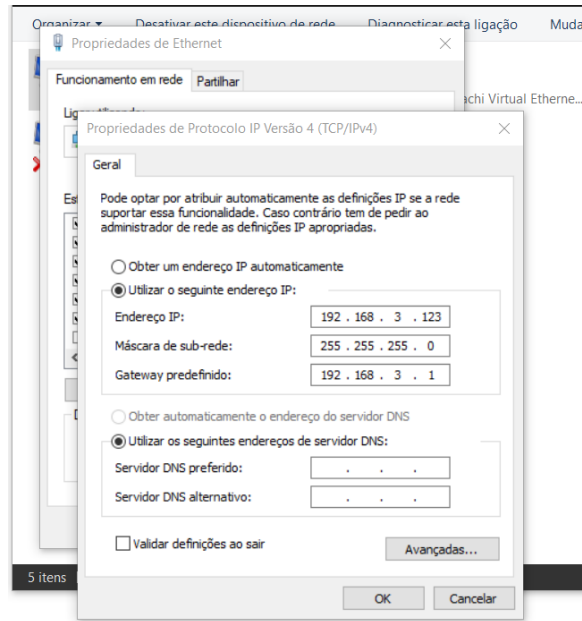


Figura 82 – Configuração do IP correto para aceder ao AGV

Depois de estabelecer a ligação correta ao AGV, basta aceder ao navegador e inserir o IP genérico para as páginas dos *dataloggers* da ASTI, que é “192.168.3.37”, conforme demonstrado na Figura 83.

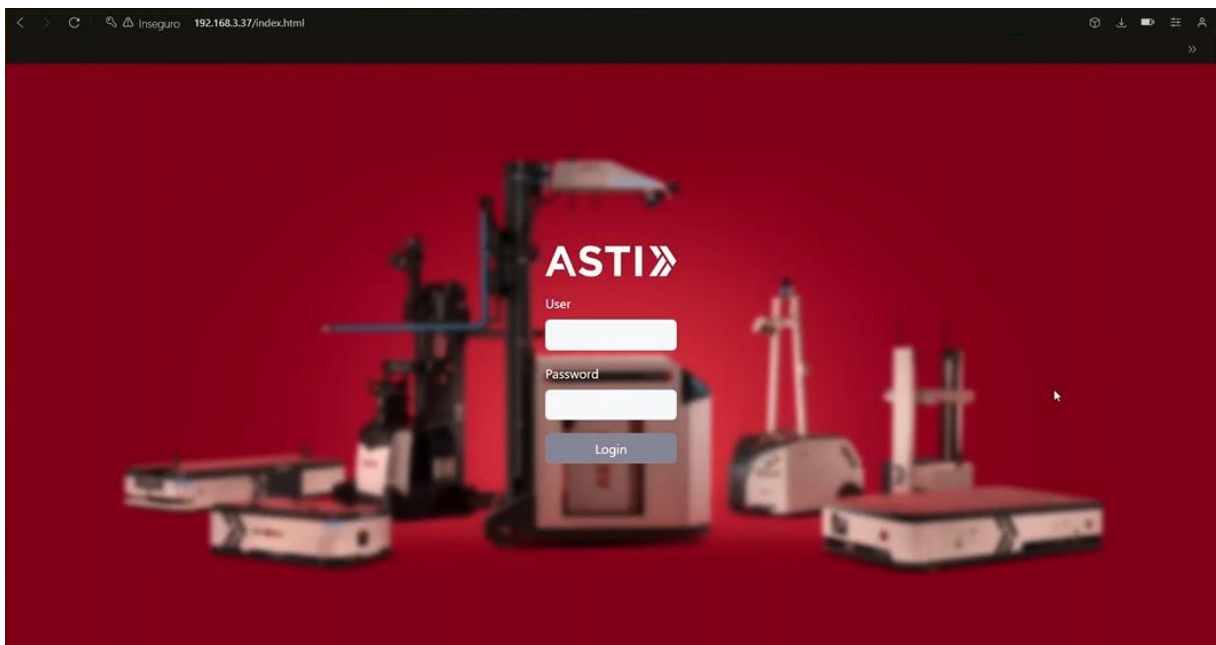


Figura 83 – Página inicial dos dataloggers da ASTI

Após realizar o login na página web dos *dataloggers*, é apresentada uma série de menus e opções de configuração. Neste momento, o mais importante é a informação em tempo real dos AGVs, como a que pode ser visualizada na Figura 84.

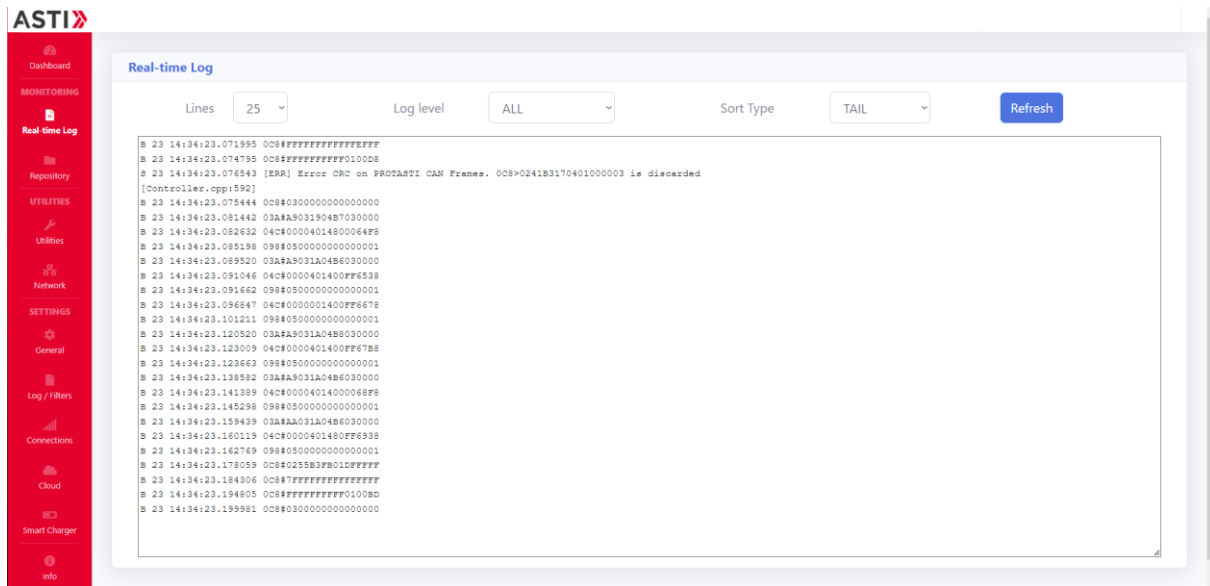


Figura 84 – Janela do Real-Time Log da página da ASTI

Todos estes dados foram recolhidos em documentos de formato de texto para serem processados, após uma filtragem, uma vez que estes registos apresentam muitas informações irrelevantes para o que é realmente o objetivo principal do sistema de supervisão. Na imagem seguinte (Figura 85), é possível observar dados importantes e necessários, como o código do display, a posição do AGV, o número da última tag, o ponto e o segmento em que este se encontra. Com toda esta informação reunida e organizada, será possível avançar para o passo seguinte.


16 12:57:11.851833 [DBG] CanDictionary	Update item: tag	Value: 20	[CanDictionary.cpp:203]
16 12:57:11.853077 [DBG] CanDictionary	Update item: point	Value: 20	[CanDictionary.cpp:203]
16 12:57:11.853823 [DBG] CanDictionary	Update item: segment	Value: 20	[CanDictionary.cpp:323]
16 12:57:11.875062 [DBG] CanDictionary	Update item: display	Value: 8	[CanDictionary.cpp:162]
16 12:57:13.416639 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0255B39A015FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF0100DC is discarded			[Controller.cpp:592]
16 12:57:13.595749 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0241B37E32000000000000010210000FFFFFFFFFFFFFFFF0000000102100000000000014210000F7FFFFFFFFFFFFFFFFFFFFFFFF0100D			[Controller.cpp:592]
16 12:57:13.748980 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0241B3970A0000000000010034006A03 is discarded			[Controller.cpp:592]
16 12:57:13.877491 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0241B38E0A000000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF0100D803 is discarded			[Controller.cpp:592]
16 12:57:13.999333 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0241B3914100000000000001421000000000001021000000000010034005D03 is discarded			[Controller.cpp:592]
16 12:57:14.153344 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0241B39641000000FFFFFFFF01005A03 is discarded			[Controller.cpp:592]
16 12:57:14.570391 [DBG] CanDictionary	Update item: display	Value: 59	[CanDictionary.cpp:162]
16 12:57:17.863081 [DBG] CanDictionary	Update item: tag	Value: 30	[CanDictionary.cpp:203]
16 12:57:17.863892 [DBG] CanDictionary	Update item: point	Value: 30	[CanDictionary.cpp:203]
16 12:57:17.865158 [DBG] CanDictionary	Update item: segment	Value: 30	[CanDictionary.cpp:323]
16 12:57:18.396109 [DBG] CanDictionary	Update item: display	Value: 8	[CanDictionary.cpp:162]
16 12:57:18.600650 [DBG] TCP Client n.1 Attempt connection to Server: 10.7.48.254 (10.7.48.254) Port:4660			[TCPIPManager.cpp:526]
16 12:57:20.203819 [ERR] Error CRC on PROTASTI CAN Frames. 0C8>0255B393015FFFFFFFFFFFFFFFFFFFFFFFF03 is discarded			[Controller.cpp:592]

Figura 85 – Dados recolhidos do AGV pelo datalogger

Para que a recolha dos dados seja feita de forma automática, é necessário aceder à página web do *datalogger* e extrair a informação, conforme demonstrado anteriormente. Para conseguir realizar esta tarefa de forma automática e em simultâneo para vários AGVs, foi necessário criar um programa capaz de realizar as ações que um operador faz, ou seja, aceder à página do *datalogger* pelo IP do respetivo AGV que queremos monitorizar, efetuar o login na página da ASTI, aceder à página dos *logs* e recolher os dados para posterior processamento.

Para realizar esta tarefa, foi utilizada a linguagem Python em conjunto com o pacote Selenium. Este é um pacote utilizado para automatizar a interação com o navegador de Internet a partir do Python. Este pacote é capaz de recolher as informações necessárias repetidamente, sem ser necessária a intervenção de um operador para realizar tal tarefa repetitiva. Os dados recolhidos serão armazenados num ficheiro de log. Apesar da própria página de *logs* da ASTI apresentar um filtro para algumas informações pouco relevantes, existem sempre alguns dados irrelevantes que acabam por passar para o ficheiro. Desta forma, foi desenvolvido um novo script em Python para abrir o ficheiro criado a partir das páginas de cada *datalogger* e processá-lo, de modo a reter apenas a informação relevante, conforme ilustrado na Figura 86, e armazená-la num ficheiro, que posteriormente será processado para uma base de dados.

```
2024-01-18 12:10:47,689 - IP: 192.168.0.105 - Valor: 16 12:57:55.504034 [DBG] CanDictionary - Update item: tag Value: 11
2024-01-18 12:10:47,925 - IP: 192.168.0.102 - Valor: 16 12:57:55.804928 [DBG] CanDictionary - Update item: tag Value: 7
2024-01-18 12:10:47,956 - IP: 192.168.0.104 - Valor: 16 12:57:55.504034 [DBG] CanDictionary - Update item: tag Value: 11
2024-01-18 12:10:56,766 - Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:10:57,002 - Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:10:57,034 - Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:00,822 - Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:01,068 - Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:01,098 - Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:04,903 - Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:05,150 - Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken
2024-01-18 12:11:05,181 - Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken
```



```
IP: 192.168.0.105 -> Value: 11
IP: 192.168.0.102 -> Value: 7
IP: 192.168.0.104 -> Value: 11
```

Figura 86 – Dados retirados da página da ASTI processados

Caso esta opção não seja exequível, existe ainda uma terceira alternativa, que consiste no desenvolvimento de hardware específico para recolher a informação necessária. No entanto, esta opção será mais dispendiosa visto ser necessário desenvolver hardware dedicado para cada modelo de AGV existente.

Assim, o próximo objetivo é recolher os dados através da base de dados da ASTI, presente em cada *data logger*. No entanto, o problema reside no facto de ser necessária uma licença e um software próprio da marca, para aceder a esta base de dados. Embora tenha sido possível recolher alguma informação sem recorrer a estes serviços, todos os dados obtidos estavam encriptados. Em suma, sem o suporte da própria marca não será possível avançar nesta fase do processo.

Foi então necessário adquirir e instalar uma licença do software SIGAT MultiAGV no servidor anteriormente mencionado, que tem como função principal executar todo o software em desenvolvimento, bem como armazenar as bases de dados.

A instalação do software SIGAT MultiAGV foi realizada com a assistência de um representante da ABB/ASTI, a empresa responsável pela produção dos AGVs. Após a instalação do mesmo, iniciaram-se os procedimentos de teste e análise do referido software. A interface deste programa é apresentada na Figura 89.

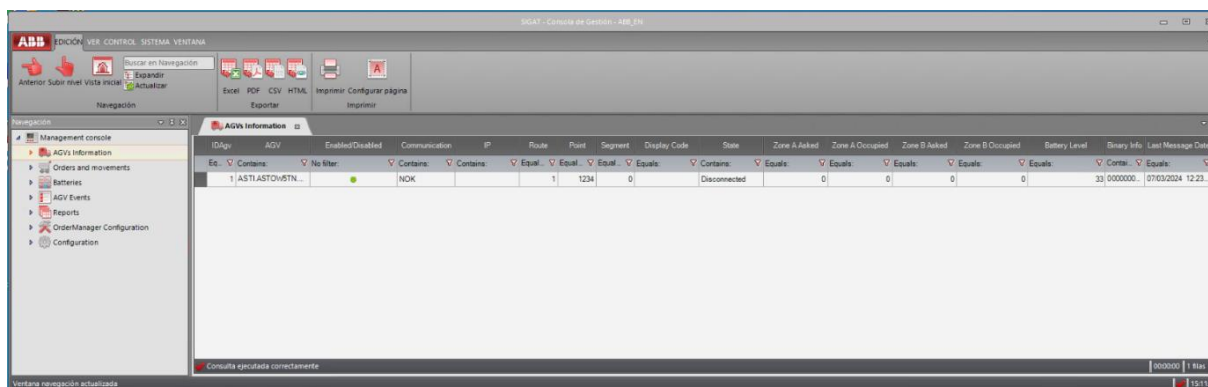


Figura 89 – Interface do programa SIGAT MultiAGV

Após a instalação completa do software, foi realizada uma pesquisa intensiva para localizar as informações mais relevantes, como os tipos de erros, filtrar por diferentes línguas, erros do servidor e dos AGVs, entre outras muitas opções e tabelas. Para que esta pesquisa fosse realizada de forma mais prática, foi utilizado o software da Microsoft, o SQL Server Management Studio, já mencionado anteriormente. Ao ligar este programa à base de dados do SIGAT MultiAGV, foi possível verificar a existência de inúmeras tabelas, conforme ilustrado na Figura 90.

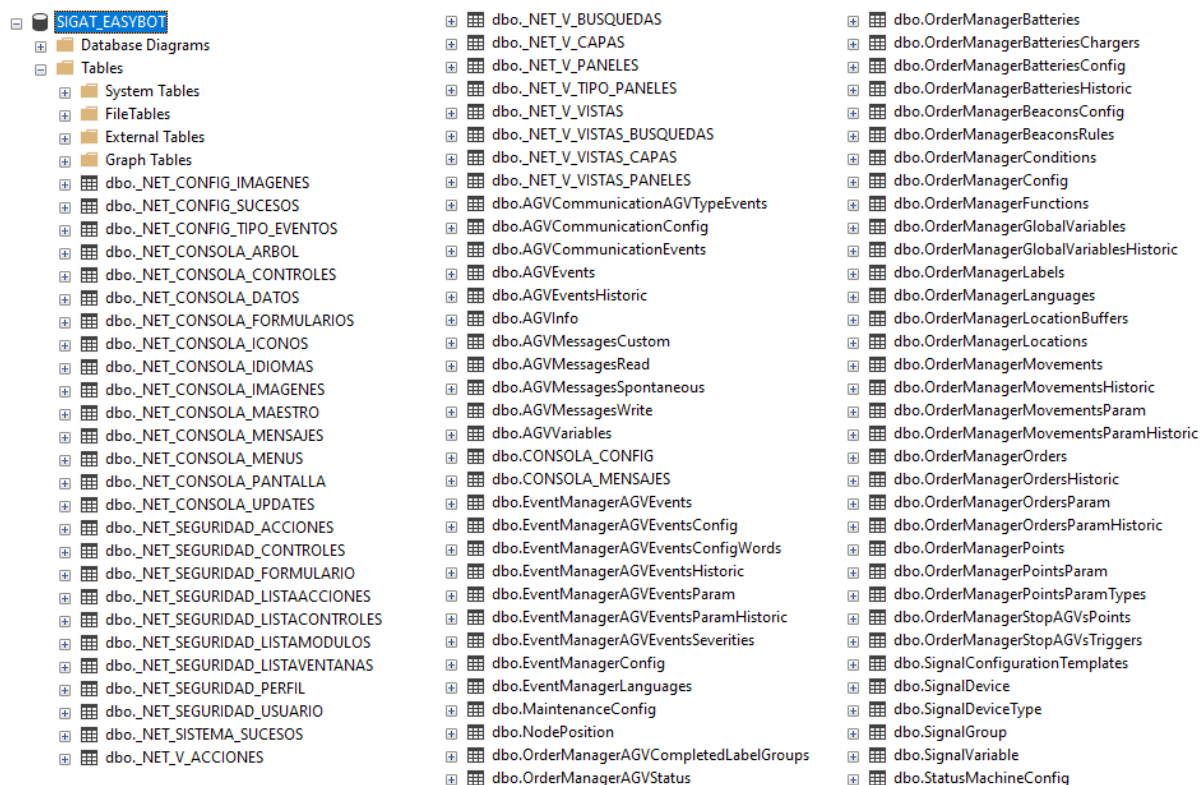


Figura 90 – Tabelas existentes na base de dados do SIGAT MultiAGV

Com o software já em funcionamento e a receber todas as informações relativas ao AGV, foram realizados diversos testes, incluindo testes de movimentação, de limites, de ângulos de mudança de direção, de deteção de obstáculos e a simulação de diferentes erros. O objetivo destes testes era observar as reações do sistema e, conseqüentemente, as atualizações registadas na base de dados.

Tornou-se então necessário efetuar uma análise detalhada à base de dados. Após a análise, foram executadas consultas/*queries* em SQL, que permitiram observar as informações mais relevantes.

As *queries* são comandos ou instruções que permitem consultar e manipular dados numa base de dados. Através das *queries*, é possível filtrar e organizar dados, o que facilita a análise das informações armazenadas.

No contexto deste projeto, a *query* principal, Figura 91, foi executada com o objetivo de extrair as informações mais relevantes sobre o estado dos AGVs. Entre os dados obtidos estão: o ID, o nome, o nível de bateria, o IP, a rota, o ponto, o segmento, o estado, a data da última posição, o número de erros, a lista atual de erros e a data da última atualização dessa mesma lista.

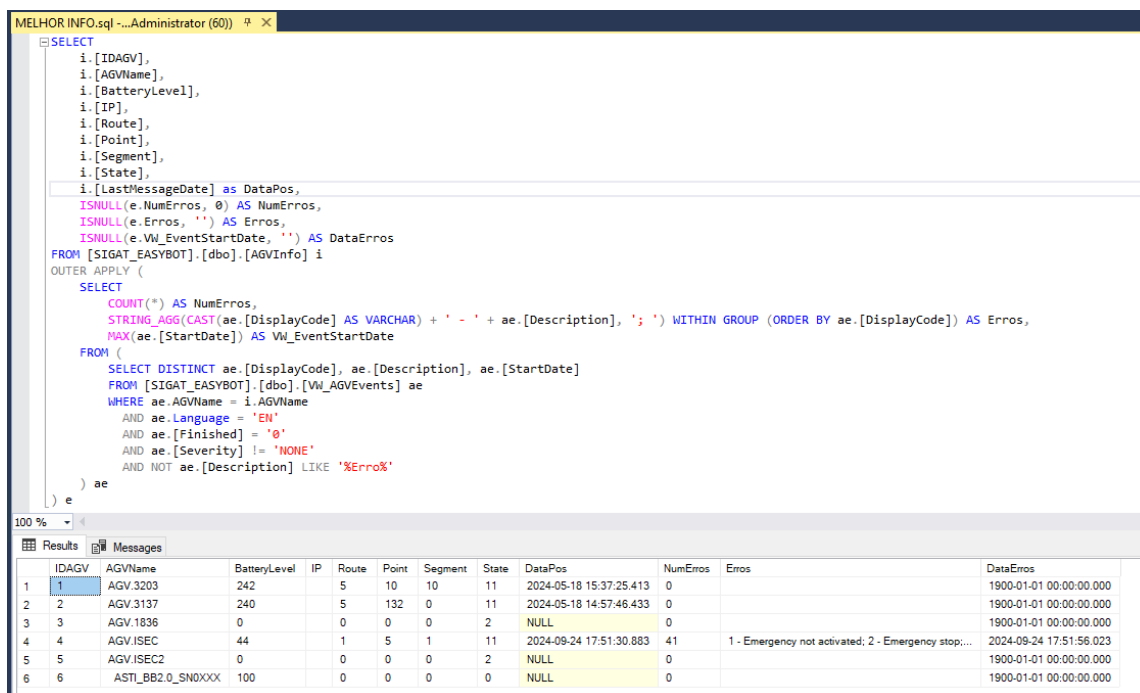


Figura 91 – Query principal com a informação relevante resumida

3.7 – Definição da base de dados para o software em desenvolvimento

A base de dados desenvolvida para o software de supervisão, ilustrada na Figura 92, foi elaborada com a linguagem SQL presente no ficheiro Anexo G – DB_Projeto_Tese.sql, Anexo G, recorrendo ao software XAMPP para criar um servidor de PHPMyAdmin. Esta base de dados é composta por três pilares principais interligados entre si. O primeiro pilar consiste na construção de uma tabela para os utilizadores, onde estão definidos diferentes níveis de acesso. Desta forma, operadores com distintos níveis de acesso podem realizar operações com maior ou menor impacto no sistema.

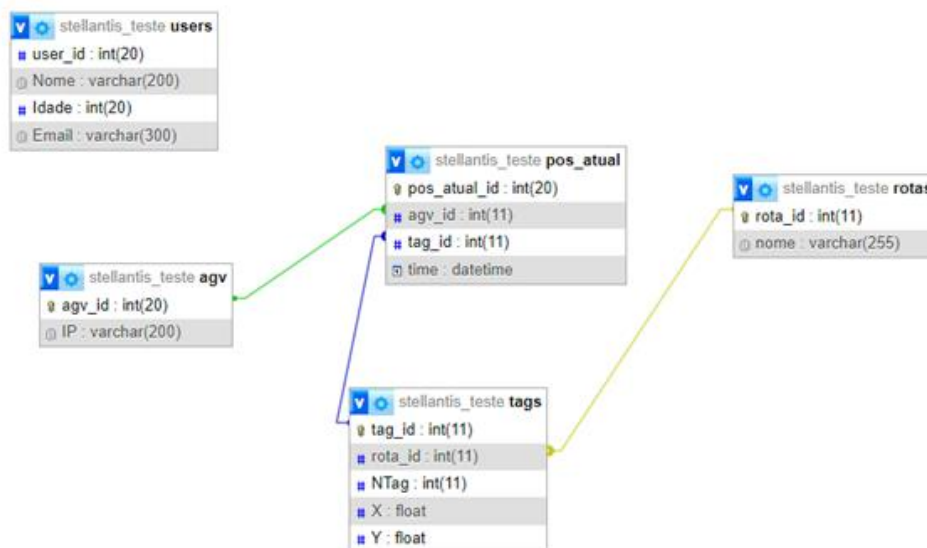


Figura 92 – Base de dados para o software em desenvolvimento

A base de dados será também responsável por armazenar toda a informação relativa às rotas e tags, garantindo assim a organização e o correto funcionamento do software. Como ilustrado na Figura 93, foi criada uma tabela específica com todos os nomes de rotas existentes, cada um associado a um ID respetivo. Além disso, existe também uma outra tabela distinta onde são guardadas todas as tags existentes na unidade fabril.

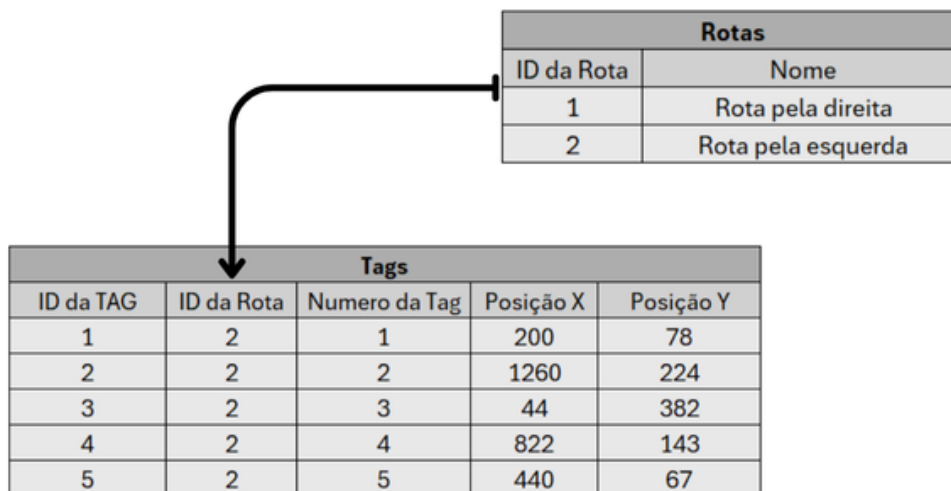


Figura 93 – Excerto um da base de dados desenvolvida

Além das rotas, é igualmente necessário guardar a informação relativa aos AGVs na base de dados. Para esse efeito, foram criadas duas tabelas, conforme ilustrado na Figura 94. Uma destas tabelas destina-se a guardar a informação que identifica cada AGV. Nesta versão é utilizado o endereço IP, mas que no futuro serão incluídos outros métodos, como, por exemplo, a matrícula. A outra tabela resume a posição mais recente de cada AGV e a partir desta, é possível consultar o tempo da última atualização de posição, identificar qual o AGV que se movimentou, e, através da consulta da tag, determinar as coordenadas X e Y da última posição publicada pelo AGV.

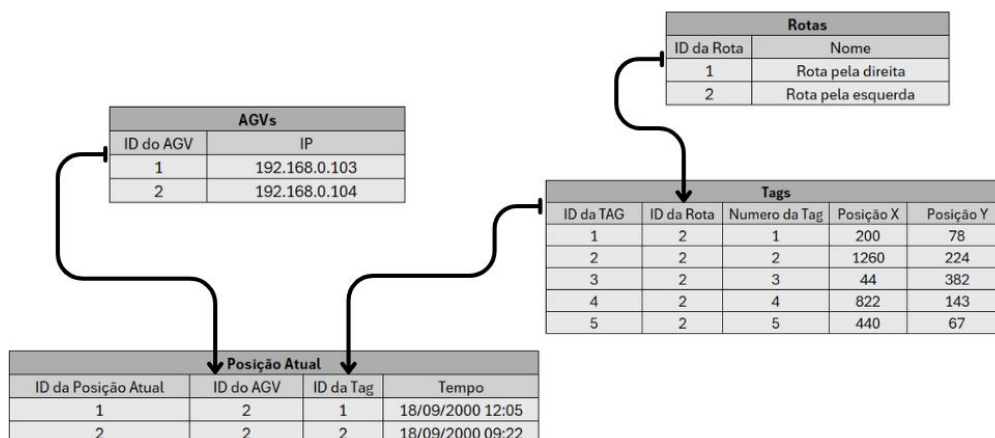


Figura 94 – Excerto dois da base de dados desenvolvida

Uma das principais funções desta aplicação web é supervisionar a rota dos AGVs. Para detetar uma falha no circuito dos AGVs, é necessário armazenar a informação referente a todo o trajeto predefinido de cada AGV. Para este efeito, foram criadas mais 10 tabelas, cada uma com os dados sobre as respetivas rotas, informação essa retirada diretamente do software utilizado para a programação dos AGVs, o SIGAT.

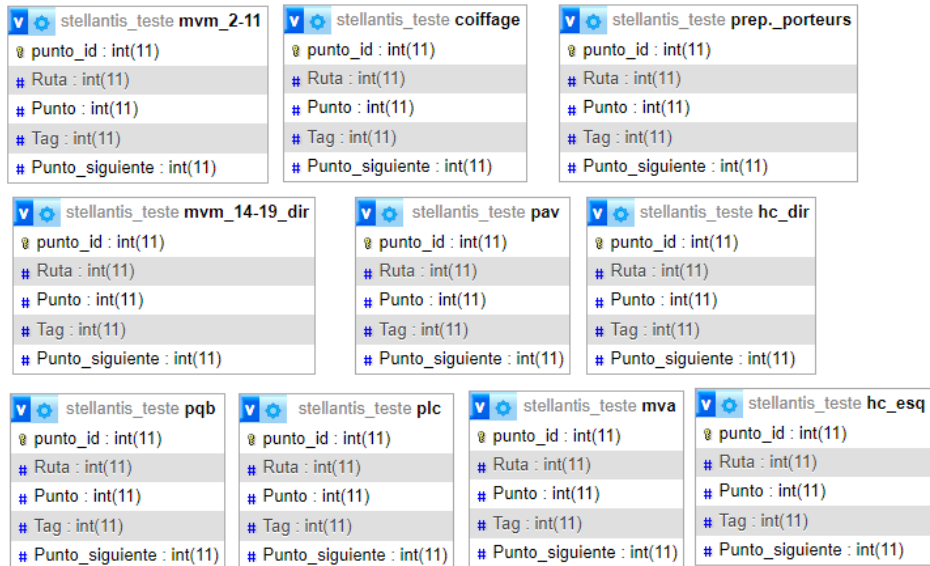


Figura 95 – Tabelas com programação dos AGVs

Em cada uma destas tabelas, são armazenados dados relativos às rotas, como o ID da rota, o ponto, a tag e o ponto seguinte. Com base nesta informação, é possível realizar uma análise para verificar se o AGV está a seguir o trajeto previsto durante o seu movimento. Por exemplo, como mostra a Figura 96, é possível verificar, em tempo real, se o AGV passa corretamente do ponto 5 para o ponto 6, comparando a tabela e os dados extraídos.

HC-DIR			
Ruta	Punto	Tag	Punto siguiente
3	4	208	300
3	5	8	6
3	6	3	63
3	10	308	11
3	11	308	33
3	23	23	108
3	31	308	10
3	32	158	31
3	33	83	90
3	34	181	35
3	35	153	103
3	36	35	37
3	37	245	110
3	38	1204	95
3	39	10	122
3	40	616	41

Figura 96 – Exemplo da informação das tabelas

Posteriormente, estes dados serão processados por um script de verificação.

3.8 – Desenvolvimento da plataforma de monitorização e de supervisão

Com a estrutura da base de dados e os dados necessários para o funcionamento do software de monitorização, iniciou-se o desenvolvimento da aplicação web. **Este desenvolvimento será realizado em várias etapas, uma vez que a maior parte do trabalho e da complexidade estão concentrados nesta fase.**

3.8.1 – Página de Login

Como mencionado anteriormente, a plataforma web incluirá diferentes níveis de acesso. Por esta razão, o desenvolvimento teve início com a criação de uma página de login (Figura 97), que atribui os respetivos níveis de acesso aos utilizadores com base nas credenciais fornecidas. Esta abordagem assegura que cada utilizador tenha acesso apenas às funcionalidades e informações que são relevantes para o seu papel na plataforma, garantindo a segurança e a integridade dos dados.

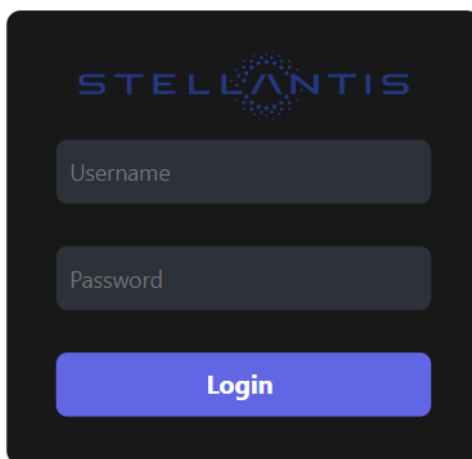


Figura 97 – Página de login da página web

O código para a criação da página de login foi escrito em HTML, CSS e PHP conforme presente no Anexo A – Index.php. Este código está dividido em duas partes principais: a estrutura da página e a lógica subjacente à mesma. A estrutura da página é definida recorrendo ao HTML, onde são estabelecidos os elementos básicos da interface, como os campos de introdução do *Username*, da *Password* e do botão de envio.

```
<body>
11.     <section class="area-login">
12.         <div class="login">
13.             <div>
14.                 
15.             </div>
16.             <form action="" method="post">
17.                 <input type="text" name="username" placeholder="Username">
18.                 <input type="password" name="password" placeholder="Password">
19.                 <input type="submit" name="login" value=Login>
20.             </form>
21.         </div>
22.     </section>
23. </body>
```

A estética da página é definida com o uso do CSS, sendo incorporado na página com a seguinte linha de código <link rel="stylesheet" href="css/style.css?v=<?php=\$version?>">. Este ficheiro CSS define o tema visual da página, incluindo o layout, as cores, as fontes e o design dos botões, garantindo uma interface agradável e intuitiva para o utilizador. Sem o CSS, a página seria funcional, mas teria um aspeto básico e pouco perceptível, dificultando a experiência de utilização.

Com o HTML e o CSS a funcionar corretamente e em sintonia é possível exibir uma página visualmente agradável, de fácil compreensão e confortável para o utilizador.

No entanto, apesar de tudo parecer visualmente correto, esta página não consegue realizar nenhuma ação sem recorrer a outro tipo de programação no lado do servidor, como por exemplo o uso do código PHP apresentado. Neste caso, a página web utiliza PHP para verificar se os campos estão preenchidos ao pressionar o botão de login. Se estes estiverem preenchidos, é feita uma consulta à base de dados em MySQL para verificar se os dados correspondem à tabela de utilizadores previamente definida. Se os dados do utilizador e a palavra-passe coincidirem, o código em PHP redireciona o utilizador para a página principal da plataforma web. Se algum dos dados não for validado, o processo não avança e é exibida uma mensagem de "credenciais erradas", uma vez que é requerido um login válido para aceder à página principal da plataforma.

```
/* CASO O BOTÃO LOGIN SEJA PREMIDO*/
33. if(isset($_POST['login'])){
34.     $_SESSION['username'] = $_POST['username'];
35.     $_SESSION['password'] = $_POST['password'];
36.
37.     /* PARA VERIFICAR DADOS DE LOGIN! */
38.     $sql_loginusers = "SELECT *
39.     FROM `users`
40.     WHERE `Nome` LIKE '$_SESSION[username]'
41.     AND `Idade` LIKE '$_SESSION[password]'";
42.
43.     $sql_query = mysqli_query($con, $sql_loginusers);
44.
45.     if ($sql_query->num_rows != 0) {
46.         $_SESSION['user'] = true;
47.         header("Location: pagina.html");
48.     }else{
49.         echo "Credenciais erradas!";
50.     }
}
```

3.8.2 – Edição de rotas e tags

Para simplificar e facilitar o processo de realização de vários testes, optou-se por criar uma página semelhante à ilustrada na Figura 98, cujo código está descrito no Anexo B – Pagina.html. Nesta página, foi implementada uma barra de navegação na parte superior (cujo código se encontra no Anexo C – Topnav.html). No futuro, esta barra será utilizada para adicionar diversas opções de funcionalidades que contribuirão para a estruturação e organização da aplicação em desenvolvimento. Inicialmente, esta barra de navegação inclui, dois botões: um que permite selecionar entre diferentes rotas e outro que possibilita ao utilizador terminar a sessão. O código responsável por esta última funcionalidade está detalhado no Anexo D – Logout.php.

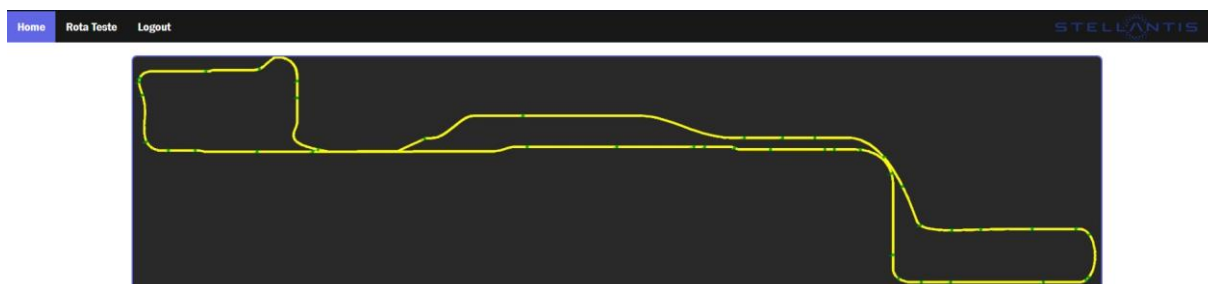


Figura 98 – Página web versão 1

As informações sobre as rotas da unidade fabril, mencionadas no tópico 3.3, incluem detalhes como o nome das rotas e dados das respetivas tags, tais como o número da tag e as suas posições X e Y. Inicialmente, estes dados são armazenados e atualizados num ficheiro em formato “. cad” que é utilizado no AutoCAD. No entanto, é necessário armazenar estas informações numa base de dados, facilitando assim o avanço no desenvolvimento do software de monitorização.

Para simplificar o processo de armazenamento das posições das tags de cada rota, foi extraída uma imagem da rota do AutoCAD e integrada na página web dentro de um elemento Canvas. O Canvas, é um elemento gráfico que possibilita diversas operações, como animações e interatividade com a página.

A solução desenvolvida para introduzir os dados consistiu na criação de um código capaz de detetar a posição onde o operador clica com o rato, registando as coordenadas dessa posição. Desta forma, o operador utiliza a rota retirada do AutoCAD, onde os pontos são visíveis (conforme ilustrado na Figura 98), e clica com o rato em cima destes, armazenando as respetivas coordenadas X e Y.

O principal desafio foi garantir que toda a lógica pretendida funcionasse corretamente com a linguagem de programação JavaScript. Esta fase exigiu um processo de tentativa e erro significativo o que consumiu bastante tempo até que a página web funcionasse da forma esperada, combinando o JavaScript com outras linguagens fundamentais à estrutura correta de uma página web, como o HTML, o CSS e o PHP.

Além disso, o código desenvolvido permite alternar entre as diferentes rotas existentes na fábrica da Stellantis, alterando de forma dinâmica a imagem de fundo exibida no Canvas, conforme a rota selecionada pelo operador na página web.

Uma das funcionalidades mais desafiantes implementadas em JS foi a utilização da interface de programação de aplicações (API) chamada Fetch. O Fetch é utilizado para estabelecer uma comunicação assíncrona entre o *frontend* (JS) e o *backend* (PHP) da página web, permitindo que as alterações feitas pelo utilizador sejam enviadas para o servidor sem a necessidade de atualizar a página.

A lógica por de trás de todo o processo começa quando o utilizador altera ou adiciona informações à página web, como por exemplo quando o operador pretende alterar a rota que deseja visualizar. Estas alterações são recolhidas pelo JS, que organiza toda a informação no formato adequado. Em seguida, a API Fetch envia estes dados para o código em PHP através de uma requisição HTTP, do tipo POST. O servidor (PHP) processa os dados recebidos, realiza as operações necessárias na base de dados, como inserir, atualizar ou remover dados, e devolve uma resposta ao cliente (JS). Após receber a resposta do lado do servidor, o Fetch interpreta os dados e executa as respetivas funções em JS que vão modificar a página web do utilizador, sem ser necessário de recarregar a página, conforme ilustrado na Figura 99.

Os excertos do código responsável por esta comunicação entre o cliente e o servidor (JS e PHP) pode ser consultado no Anexo E – Sistema.php e no Anexo F – Script.js.

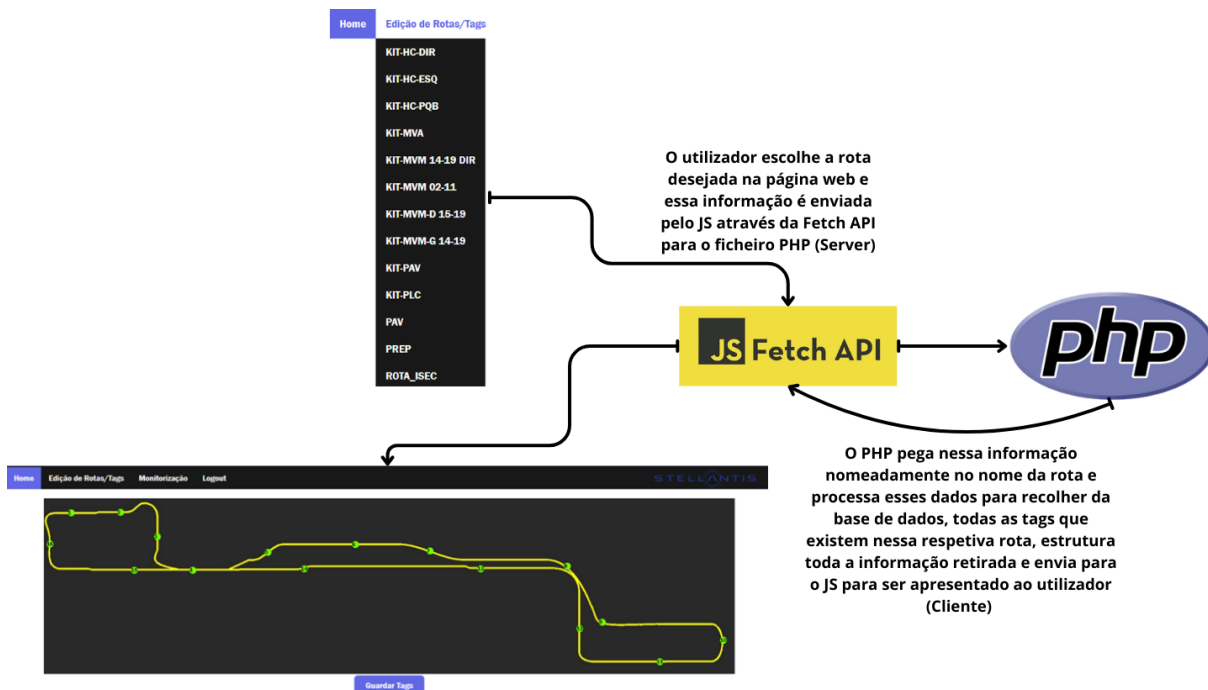


Figura 99 – Lógica da mudança de rota na página web

Com a comunicação do Fetch a funcionar corretamente, o utilizador pode realizar diversas ações. É possível adicionar novas tags ou editar as existentes movendo-as

diretamente sobre a rota com o auxílio do rato. Além disso, os valores das tags podem ser alterados diretamente na tabela de tags, preenchendo os campos disponíveis com os valores pretendidos como o número de tag e as suas posições X e Y. Caso necessário, também é possível remover as tags, como ilustrado na Figura 100.

Após posicionar as tags nos locais desejados é possível guardar as mudanças efetuadas utilizando o botão “Guardar Tags”, para que os valores das mesmas sejam armazenados na base de dados.

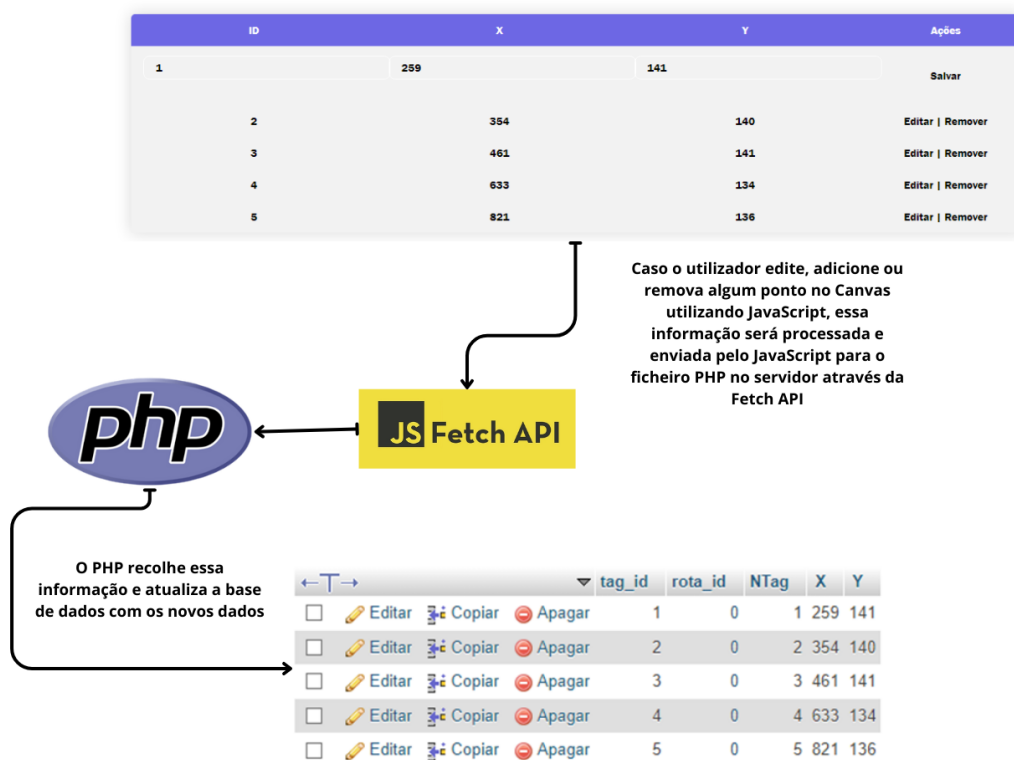


Figura 100 – Lógica de guardar tags na página web

3.8.3 – Posição atual dos AGVs

Conforme explicado no ponto 3.6, os dados relativos aos AGVs são extraídos do *datalogger* e processados até que a informação necessária seja consolidada na base de dados. Agora que a base de dados está devidamente definida e estruturada, o

passo seguinte compreende a integração destes dados processados na base de dados, conforme podemos observar na Figura 101.

Para este efeito, foi desenvolvido um script em Python, capaz de inserir estes dados na base de dados utilizando o seguinte comando SQL:

```
1. INSERT INTO pos_atual (agv_id, tag_id, time) \
2. SELECT agv.agv_id, tags.tag_id, NOW() \
3. FROM agv JOIN tags ON agv.IP = %s AND tags.NTag = %s;"
```

Este comando executa várias operações, por exemplo, ao processar dados como “IP: 192.168.0.104 -> Value: 11”, o script verifica na tabela ‘agv’ qual é o AGV que está associado ao IP ‘192.168.0.104’ e na tabela tags, qual é a tag que está associada ao valor ‘11’. Com base nestas associações, o comando insere ‘agv_id’ e ‘tag_id’ na tabela ‘pos_atual’, juntamente com a data e hora em que a inserção foi realizada.

Com esta abordagem, as operações que dependem destes dados podem ser realizadas com informações sempre atualizadas, melhorando a eficiência e a fiabilidade do sistema.

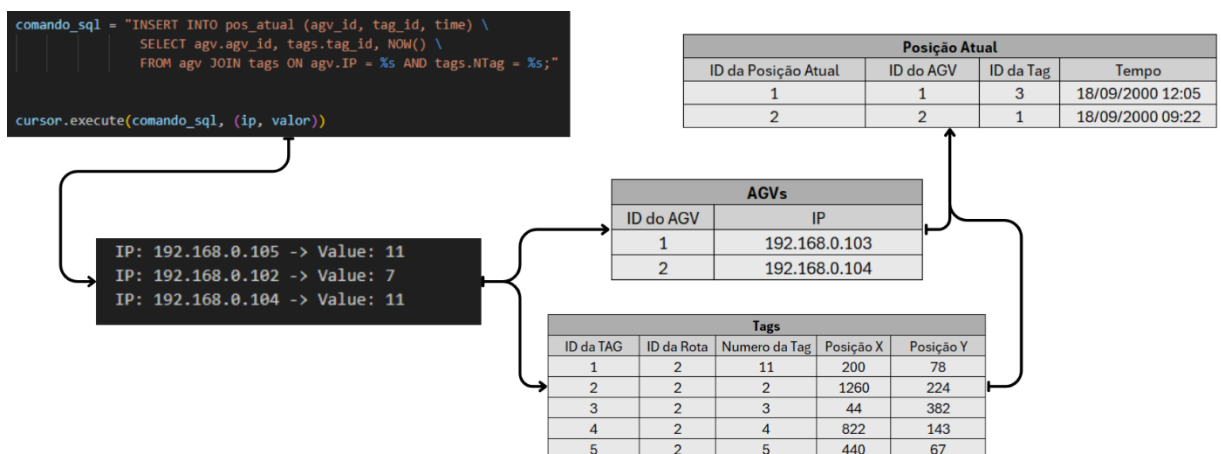


Figura 101 – Lógica para a posição do AGV na página web

Agora que possuímos todas as informações relativas à posição de cada AGV, basta criar um script em JS (Anexo F – Script.js) que executa *loop* em intervalos de tempo para atualizar a posição de cada AGV na rota, de acordo com o ID correspondente e a informação com a data mais recente. O desenho do AGV é realizado novamente com a ajuda do Canvas, ao criar um elemento posicionado sobre as coordenadas X e Y da tag em que o AGV se encontra, como se pode observar na Figura 102. Os dados necessários para esta atualização são obtidos através de consultas à base de dados.

Algumas destas anomalias podem ser consultadas diretamente pelo código apresentado pelo *display* do AGV, que está a ser posteriormente processado para a base de dados.

Para garantir uma maior robustez, foi desenvolvido um script adicional responsável por verificar se o número lido da tag está em conformidade com a tabela de programação previamente programada e introduzida tanto no AGV como na base de dados da aplicação que pode ser consultada na Tabela 3.

Ao consultar a Tabela 3, é possível observar que cada ponto possui uma tag associada e um ponto seguinte definido. Para facilitar a interpretação, é possível consultar também a Figura 103, que ilustra a rota com o ID 12. Este ID refere-se à rota implementada no ISEC, utilizada durante a realização dos testes.

Os pontos visíveis nesta rota correspondem às tags presentes na coluna “Número da tag” da Tabela 3. Por exemplo, se o AGV ler a tag 16, significa que este se encontra no ponto 3 da tabela e que o próximo ponto será o ponto 4, o que significa que a próxima tag a ser lida pelo AGV deverá ser a tag com o número 20. Esta lógica baseia-se na tabela de programação predefinida no AGV e mantém-se ao longo de toda a rota funcionando em *loop*.

O script desenvolvido compara continuamente a tag esperada com a tag efetivamente lida e, em caso de discrepância, envia um alerta imediato para o operador.

Tabela 3 – Tabela do trajeto programado da rota do ISEC

Rota ID	Ponto atual	Número da tag	Ponto seguinte
12	1	11	2
12	2	12	3
12	3	16	4
12	4	20	5
12	5	19	6
12	6	11	7
12	7	13	8
12	8	14	9
12	9	15	10
12	10	17	11
12	11	18	12
12	12	14	13
12	13	13	1

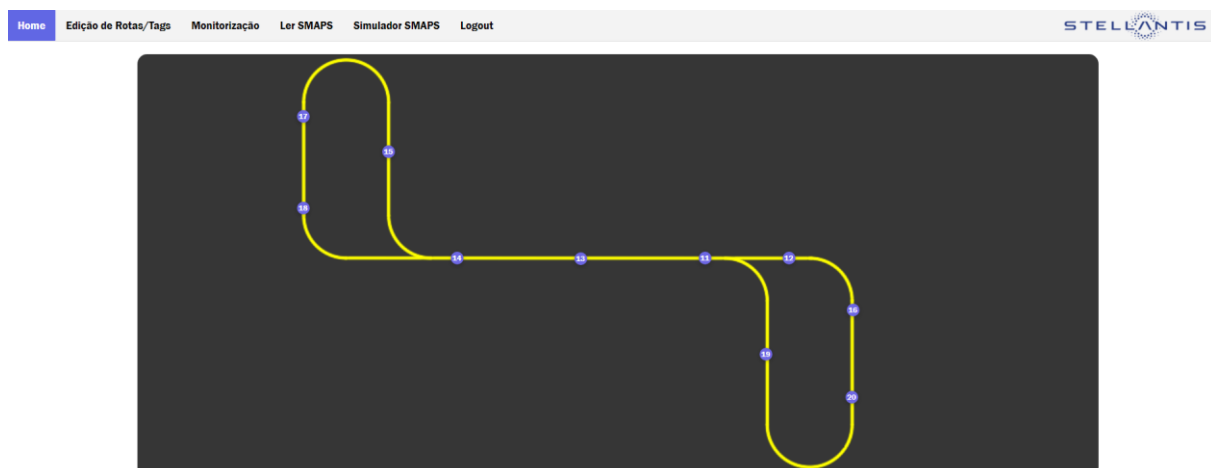


Figura 103 – Rota do ISEC correspondente ao ID 12 na base de dados

Quando um alarme é acionado, é possível seleccionar o AGV em questão e aceder a todas as informações relevantes, como o seu número identificativo, o estado atual e os detalhes das anomalias registadas. Esta funcionalidade é exemplificada na Figura 104, onde são apresentadas as informações do AGV selecionado.

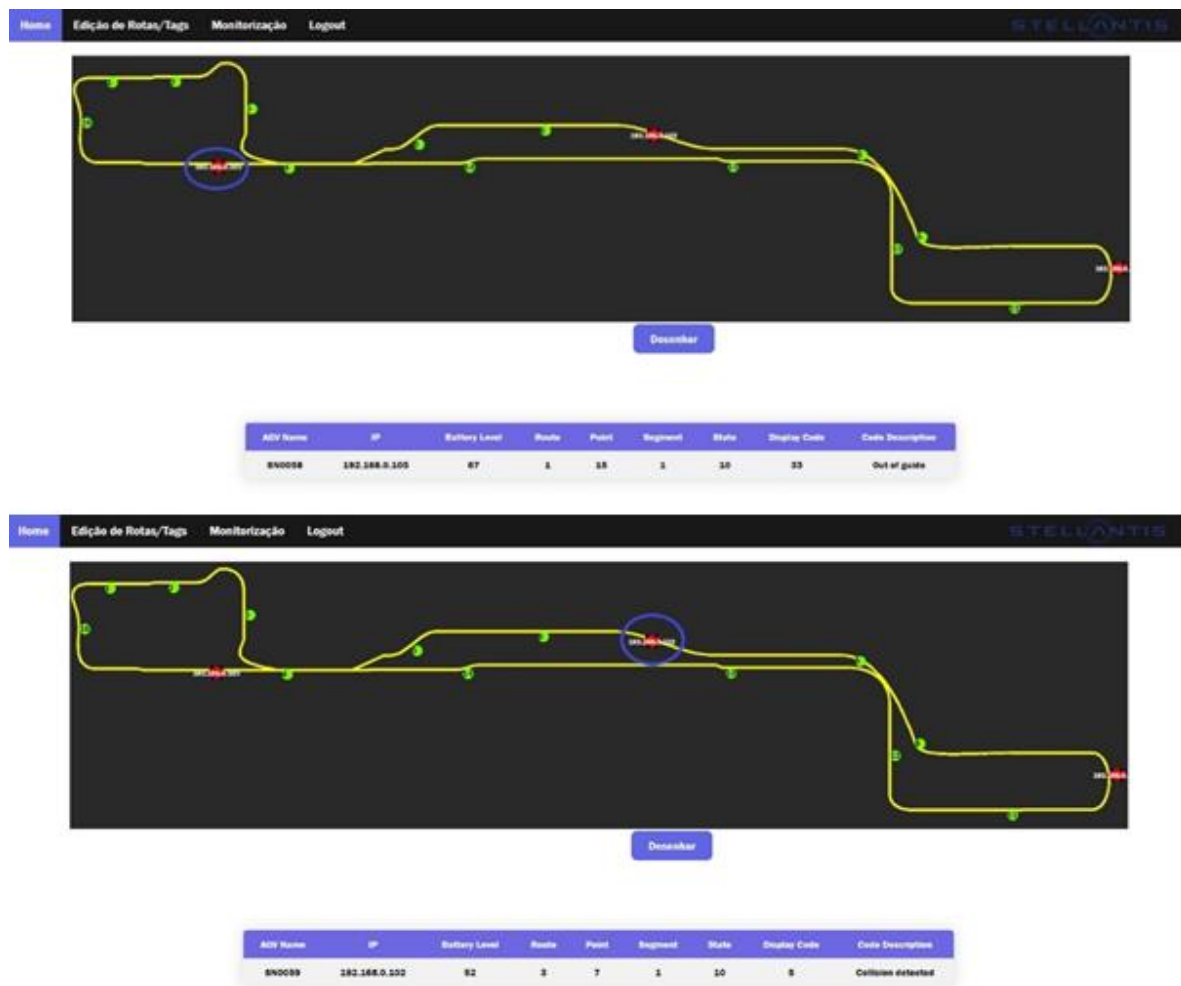


Figura 104 – Exemplo das informações relativas ao AGV selecionado

A aplicação apresenta também as informações de forma visual, de modo a alertar os utilizadores de forma intuitiva sobre o estado dos veículos. Por exemplo, as cores associadas a cada AGV indicam o seu estado atual e quaisquer eventuais códigos de erro visíveis no visor, conforme demonstrado na Tabela 4 e figuras 105 a 108.

Tabela 4 – Sinóptico das cores do estado dos AGVs

Cor	Código da cor
Vermelho	O AGV está parado, devido a uma falha ou avaria
Amarelo	O AGV está parado porque encontra um obstáculo.
Azul	O AGV está a carregar as baterias
Verde	O AGV está a funcionar corretamente



Figura 105 – AGV vermelho



Figura 106 – AGV amarelo



Figura 107 – AGV Azul



Figura 108 – AGV verde

Este sistema de cores permite aos operadores uma supervisão rápida e eficiente de toda a unidade fabril, proporcionando uma visão global, clara e organizada.

3.9 – Estado final do software de supervisão e monitorização

Por fim, é possível observar o estado final do software de supervisão e monitorização desenvolvido ao longo deste projeto. Nesta fase, a aplicação web consegue cumprir os objetivos inicialmente definidos. Para acederem à aplicação, os utilizadores devem efetuar o login na mesma, conforme ilustrado na Figura 109, onde lhes é atribuído o respetivo nível de acesso.

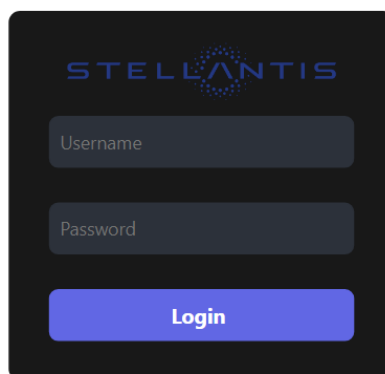


Figura 109 – Estado final da página de login

A Figura 110, apresenta a página inicial “Home” da aplicação web, que inclui uma introdução sucinta e uma explicação sobre a WP10 e o PP17 do projeto GreenAuto na qual este projeto se insere.



Figura 110 – Estado final da página “Home”

Na página de “Edição de Rotas e Tags”, ilustrada na Figura 111, é possível realizar diversas ações, como adicionar novas tags, editar tags movendo-as diretamente sobre a rota com o auxílio do rato, ou alterar os seus valores diretamente na tabela de tags, preenchendo os campos disponíveis, como o número da tag e as suas posições X e Y, com os valores pretendidos. Após efetuar as alterações, é possível guardar as mudanças efetuadas ou remover tags da rota selecionada.

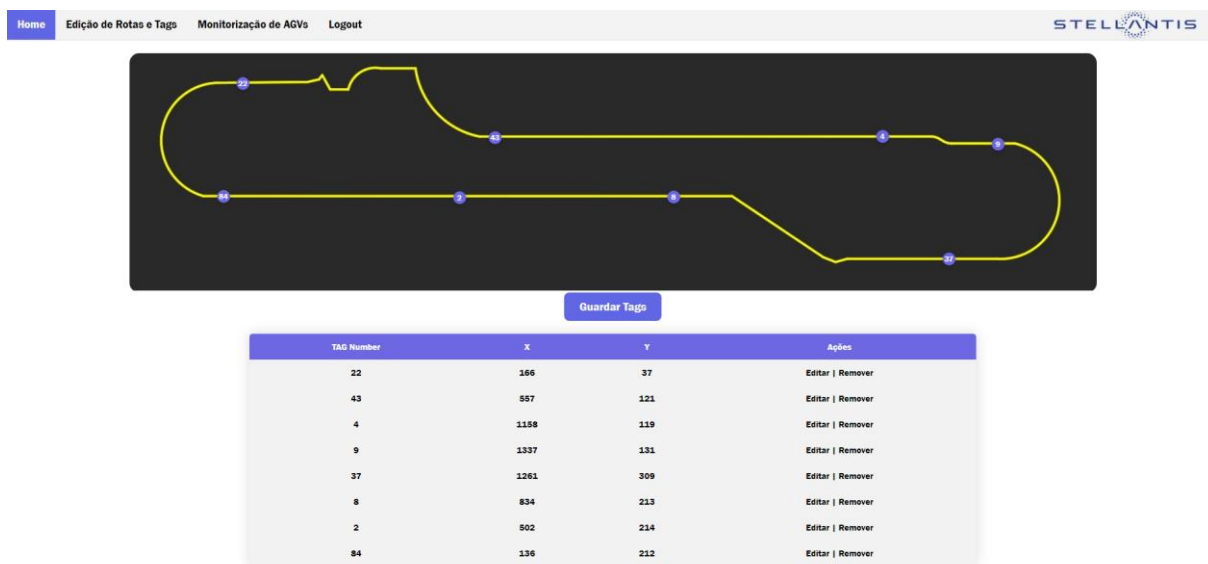


Figura 111 – Estado final da página de “Edição de Rotas e Tags”

Ao aceder à página de “Monitorização de AGVs”, conforme demonstrado na Figura 112, é possível visualizar a posição atual do AGV utilizado para testes. Ao selecionar o AGV, são exibidos os dados mais relevantes sobre o mesmo.

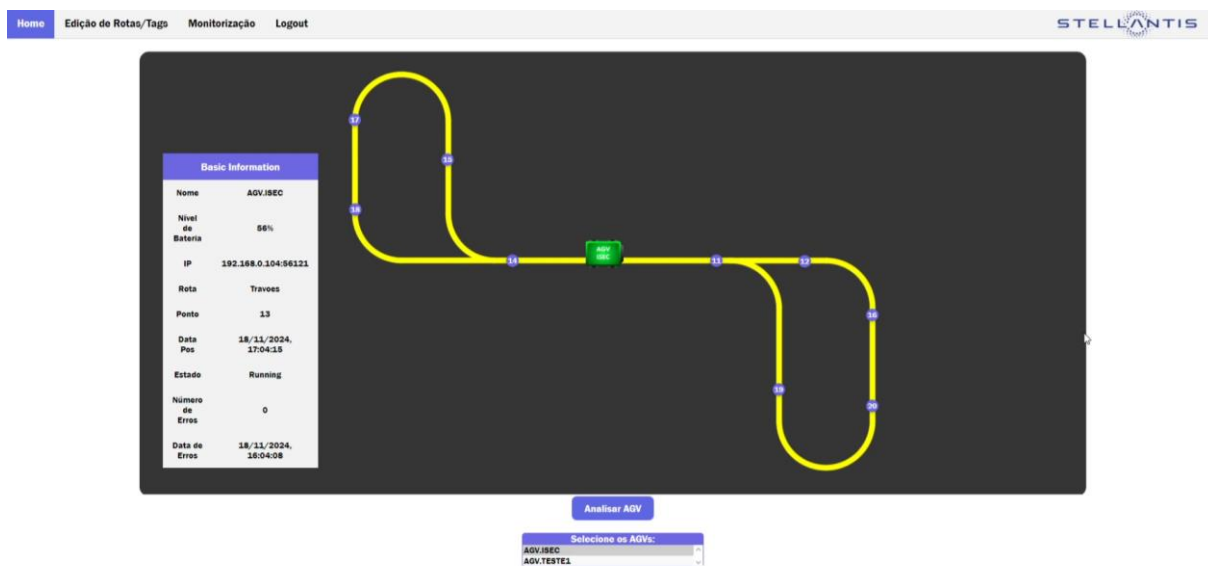


Figura 112 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Running”

Na Figura 112 e Figura 113, é possível verificar que o AGV se deslocou, o que é refletido pelos dados atualizados na tabela. Embora a posição difira, a mudança de cor do AGV tanto nessas figuras como na Figura 114 é igualmente notável. Na primeira figura, o AGV estava na cor verde, indicando estado de “Running” e sem erros. Na Figura 113, encontra-se na cor amarela, num estado de “Alert” e apresentava um erro, de anticolisão, pois o AGV detetou um obstáculo na sua rota.

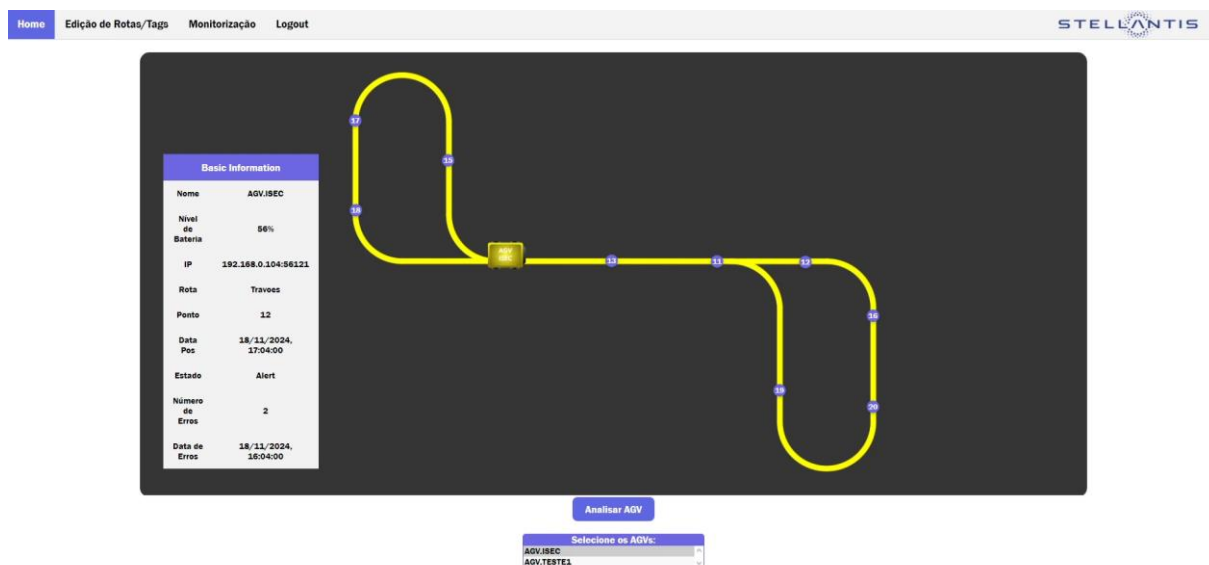


Figura 113 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Alert”

Na Figura 114, o AGV encontra-se num estado de “Emergency” e apresenta vários erros, erros estes resultantes da ativação do botão de emergência do AGV.

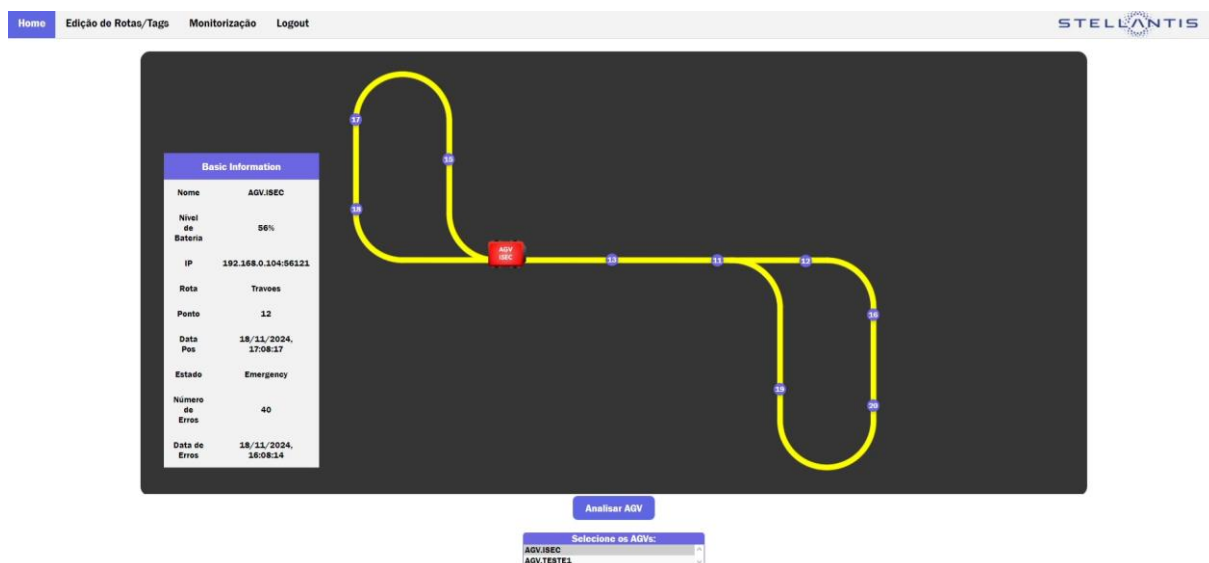


Figura 114 – Estado final da página de “Monitorização de AGVs” com o AGV em estado “Emergency”

Estas mudanças na cor do AGV seguem o esquema de cores apresentado anteriormente e fazem parte do sistema de supervisão da página web, cujo principal objetivo é facilitar a deteção rápida de potenciais erros.

CONCLUSÃO

Este capítulo apresenta as conclusões finais do projeto, os objetivos alcançados com o desenvolvimento do software de supervisão e monitorização para os AGVs, bem como as perspetivas futuras para o software.

4.1 – Objetivos alcançados e análise de resultados

Neste projeto, foi desenvolvido um software de supervisão e monitorização para os AGVs presentes na fábrica da Stellantis. Como discutido nos capítulos anteriores, o software inclui processos com alguma complexidade para a recolha, processamento e apresentação visual de dados dos AGVs aos operadores. Cada uma destas etapas inclui diversas tarefas associadas, de modo a assegurar o bom funcionamento da aplicação web na gestão de frota dos AGVs presentes na fábrica da Stellantis.

Durante o desenvolvimento, foram utilizados diversos softwares, tendo sido utilizadas as seguintes linguagens de programação: JavaScript, PHP, HTML e o Python. A experiência com JavaScript foi especialmente enriquecedora, dado que nunca tinha utilizado esta linguagem em projetos anteriores. Foi interessante compreender o seu funcionamento, especialmente no que se refere à integração com o HTML, responsável pela parte visual da página web, e com o PHP, que facilita a interligação de dados com as bases de dados. O Python, por sua vez, foi utilizado neste projeto para testar rapidamente a viabilidade das ideias de lógica antes de as transpor para JavaScript, que suportou a implementação final.

O software oferece várias funcionalidades essenciais, especialmente no que se refere à monitorização, permitindo alternar entre diferentes rotas na unidade fabril, previamente inseridas na aplicação web a partir de um ficheiro AutoCAD fornecido. A aplicação permite também adicionar, editar e remover tags associadas a estas rotas, utilizando lógica de processamento em JavaScript e PHP para gerir as operações realizadas e armazenar a informação das tags e das respetivas rotas na base de dados.

Adicionalmente, o sistema permite visualizar em tempo real os AGVs ativos em cada rota. Ao selecionar um AGV, os respetivos dados são exibidos numa tabela, proporcionando uma visão detalhada do seu estado de funcionamento. Esta funcionalidade exigiu um esforço e ajustes consideráveis durante o período de desenvolvimento deste projeto. Para assegurar o seu correto funcionamento, foi necessário recorrer ao JavaScript e ao PHP para processar os dados e interligar duas bases de dados distintas, a base de dados do software em desenvolvimento e a base de dados dos próprios AGVs.

No âmbito da supervisão, o software inclui funcionalidades adicionais, para além da monitorização, que facilitam a visualização do estado dos AGV, especialmente através de um sinóptico de cores, conforme descrito anteriormente. Este esquema de cores

associado a cada AGV permite uma interpretação visual do estado operacional, auxiliando os operadores na tomada de decisões rápidas e informadas.

Este mecanismo é especialmente útil para alertar automaticamente para desvios de trajeto ou eventuais falhas, como a deteção de colisões, ausência de tags e falhas na banda magnética, sendo os dados registados e atualizados em tempo real na base de dados. Para reforçar a gestão preventiva, foi também implementado um script que analisa a sequência de leitura das tags ao longo do percurso. Este recurso tem como objetivo identificar qualquer desvio da rota e sinalizar imediatamente o AGV no mapa, garantindo uma resposta rápida a possíveis anomalias, minimizando interrupções e contribuindo para a segurança e eficiência da unidade fabril.

Por fim, este software representa uma solução de supervisão e monitorização eficiente e escalável, contribuindo para uma gestão mais eficaz dos AGVs em ambiente industrial.

4.2 – Trabalho futuro

O trabalho futuro incluirá o desenvolvimento contínuo deste software, principalmente na interface visual da aplicação web (front end), uma vez que o foco deste projeto esteve mais orientado para a lógica funcional do sistema e não tanto na parte visual, que poderia ser mais intuitiva e agradável para o operador.

As próximas funcionalidades a implementar incluirão a capacidade de integrar no software outros tipos de marcas, com diferentes tipos de comunicação, e até novos tipos de robôs móveis, como os AMRs (uma versão mais atualizada dos AGVs, que já oferecem localização exata).

Outra funcionalidade a implementar consiste na utilização de semáforos (RFID) para gerir a movimentação de vários AGVs em pontos de cruzamento, visto existirem vários AGVs que passam no mesmo ponto mesmo em rotas distintas. Os semáforos permitirão que os AGVs parem ou avancem de acordo com a ordem emitida pelo semáforo. O objetivo é incorporar esta lógica no software em desenvolvimento, de modo que os pontos dos semáforos sejam exibidos no mapa e, ao seleccioná-los, seja possível visualizar qual é o próximo AGV a passar.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] "Agenda Green Auto, "Agenda Green Auto - Inovação Verde para a Indústria Automóvel," Agenda Green Auto [Online]. Available: <http://www.agendagreenauto.pt/>. Accessed on: Jan. 6, 2024.
- [2] Agenda Green Auto, "O Projeto - Agenda Green Auto," Agenda Green Auto. [Online]. Available: <https://www.agendagreenauto.pt/projeto/>. Accessed on: Jan. 6, 2024.
- [3] Agenda Green Auto, "Work Packages - Agenda Green Auto," Agenda Green Auto. [Online]. Available: <https://www.agendagreenauto.pt/work-packages/> /. Accessed on: Jan. 6, 2024.
- [4] A. C. Pereira and F. Romero, "A review of the meanings and the implications of the Industry 4.0 concept," Procedia Manufacturing, vol. 13, pp. 1206-1214, Jan. 2017, doi: 10.1016/j.promfg.2017.09.032.
- [5] A evolução da robótica industrial: da indústria 1.0 à indústria 4.0. [Online]. Available: <https://www.universal-robots.com/br/blog/a-evolu%C3%A3o-da-rob%C3%B3tica-industrial-da-ind%C3%BAstria-10-%C3%A0-ind%C3%BAstria-40/>. Accessed on: Jan. 6, 2024.
- [6] C. Berniche, "A Revolução Industrial e os impactos na sociedade 4.0," [Online]. Available: <https://pt.linkedin.com/pulse/revolu%C3%A7%C3%A3o-industrial-e-os-impactos-na-sociedade-40-caio-berniche> Accessed on: Jan. 14, 2024.
- [7] Indústria 4.0 a Quarta Revolução industrial. [Online]. Available: https://www.compete2020.gov.pt/destaques/detalhe/industria_4ponto0 Accessed on: Jan. 14, 2024.
- [8] G. Immerman, "Industry 4.0 Advantages and Disadvantages," MachineMetrics, [Online] Available: <https://www.machinemetrics.com/blog/industry-4-0-advantages-and-disadvantages> Accessed on: Jan. 19, 2024.
- [9] Gonzalo, "Advantages and disadvantages of industry 4.0," BECOSAN®, [Online] Available: <https://www.becosan.com/industry-4-0-the-fourth-industrial-revolution> Accessed on: Jan. 19, 2024.
- [10] V. Batista, "Indústria 4.0: Como estar preparado para essa revolução?," [Online] Available: <https://pt.linkedin.com/pulse/ind%C3%BAstria-40-como-estar-preparado-para-essa-revolu%C3%A7%C3%A3o-viviele-batista> Accessed on: Jan. 20, 2024.
- [11] Industry 4.0: The Challenges and Risks | Rockwell Automation | US, Rockwell Automation. [Online] Available <https://www.rockwellautomation.com/en-us/company/news/blogs/industry-4-0--the-challenges-and-risks.html> Accessed on: Jan. 20, 2024.

- [12] NI Business Info, "Industry 4.0 challenges and risks | nibusinessinfo.co.uk," NI Business Info. [Online] Available <https://www.nibusinessinfo.co.uk/content/industry-4.0-challenges-and-risks> Accessed on: Jan. 20, 2024.
- [13] M. Martins, "O Tempo da Robótica Colaborativa," Flow Technology, [Online] Available <https://flowtech.pt/pt/blog/tempo-robotica-colaborativa> Accessed on: Jan. 29, 2024.
- [14] A. C. Pereira and F. Romero, "A review of the meanings and the implications of the Industry 4.0 concept," Procedia Manufacturing, vol. 13, pp. 1206-1214, Jan. 2017, doi: 10.1016/j.promfg.2017.09.032.
- [15] Factory of the Future: How Industry 4.0 and AI Can Transform Manufacturing, Bain, [Online] Available <https://www.bain.com/insights/factory-of-the-future-how-industry-4-0-and-ai-can-transform-manufacturing/> Accessed on: Feb. 2, 2024.
- [16] StartUs Insights, "Top 10 Industry 4.0 Trends in 2025 | StartUs Insights," StartUs Insights, [Online] Available <https://www.startus-insights.com/innovators-guide/top-10-industry-4-0-trends-innovations-in-2021/> Accessed on: Feb. 2, 2024
- [17] IFR International Federation of Robotics, "China: Robot installations grew by 44 percent," IFR International Federation of Robotics. [Online] Available <https://ifr.org/ifr-press-releases/news/china-robot-installations-grew-by-44-percent> Accessed on: Feb. 11, 2024
- [18] F. Automation, "The Past, Present and Future of AGVs - Fred Automation," Fred AGV, [Online] Available <https://www.fredagv.com/blog/the-past-present-and-future-of-agvs> Accessed on: Feb. 11, 2024
- [19] N. Urtusol, "Robôs móveis revolucionam a indústria transformadora," Kivnon, [Online] Available <https://www.kivnon.com/pt/mobile-robots-revolutionizing-the-manufacturing-industry/> Accessed on: Feb. 12, 2024
- [20] Redação, "Robôs móveis autônomos são alternativa 4.0 para logística interna de fábricas," Inforchannel, [Online]. Available: <https://inforchannel.com.br/2020/10/28/robos-moveis-autonomos-sao-alternativa-4-0-para-logistica-interna-de-fabricas/> Accessed on: Feb. 21, 2024
- [21] F. Automation, "Worker Shortages in Manufacturing Make AGVs More Beneficial Than Ever - Fred Automation," Fred AGV, [Online] Available <https://www.fredagv.com/blog/worker-shortages-in-manufacturing-make-agvs-more-beneficial-than-ever> Accessed on: Feb. 21, 2024
- [22] Timetoast, "Automated Guided Vehicle timeline.," Timetoast Timelines, 1953. [Online]. Available <https://www.timetoast.com/timelines/automated-guided-vehicle> Accessed on: Feb. 22, 2024

- [23] A. Solomon, J. Wilck, R. Kori, and Grado Dept. of Industrial & Systems Engineering at Virginia Tech, "Introduction to Automatic Guided Vehicles," [Online]. Available: <https://tamcam.tamu.edu/courses/inen416/Handouts/AGV%20Presentation.pdf> Accessed on: Feb. 25, 2024
- [24] H. Neradilová and G. Fedorko, "The Use of Computer Simulation Methods to Reach Data for Economic Analysis of Automated Logistic Systems," Open Engineering, vol. 6, no. 1, Dec. 2016, doi: 10.1515/eng-2016-0085.
- [25] U. Andersson and Luleå University of Technology, "Laser Navigation System for Automatic Guided Vehicles: From Research Prototype to Commercial Product," Luleå University of Technology, Graphic Production, research report, 2013. [Online]. Available: <http://www.diva-portal.se/smash/get/diva2:995294/FULLTEXT01.pdf> Accessed on: Feb. 25, 2024
- [26] "9 Key Differences: AMR vs AGV - Hy-Tek Intralogistics," Hy-Tek Intralogistics. [Online]. Available <https://hy-tek.com/resources/whats-the-difference-between-amr-and-agv/> Accessed on: Jun. 20, 2024
- [27] Marques Brownlee, "Tesla Factory Tour with Elon Musk!," YouTube. Aug. 20, 2018. [Online]. Available: https://www.youtube.com/watch?v=mr9kK0_7x08 Accessed on: Mar. 1, 2024
- [28] A. M. Mendes and A. M. Mendes, "Amazon apresenta 'primeiro robô móvel totalmente autónomo' para mover cargas em armazéns," Executive Digest - a Leitura Indispensável Para Executivos, Jun. 23, 2022. [Online]. Available: <https://executivedigest.sapo.pt/noticias/amazon-apresenta-primeiro-robot-movel-totalmente-autonomo-para-mover-cargas-em-armazens/> Accessed on: Mar. 1, 2024
- [29] Solving, "AGV dictionary - Solving," Solving, [Online]. Available: <https://www.solving.com/agv-dictionary/> Accessed on: Mar. 8, 2024
- [30] R. and M. Ltd, "Global and China Automated Guided Vehicle (AGV) Industry Report, 2020-2026," Research and Markets Ltd 2024. [Online]. Available: <https://www.researchandmarkets.com/reports/5238638/global-and-china-automated-guided-vehicle-agv> Accessed on: Mar. 8, 2024
- [31] Solving, "Types of Automated Guided Vehicles - Solving," Solving, Aug. 28, 2024. [Online]. Available: <https://www.solving.com/agv-dictionary/types-of-automated-guided-vehicles/> Accessed on: Mar. 11, 2024
- [32] "ASTI Mobile Robotics," Next Generation IoT, Jun. 01, 2023. [Online]. Available: <https://ngiot.eu/asti-mobile-robotics/#1638802538755-3d54710d-736f> Accessed on: Mar. 11, 2024

- [33] C. Benevides, "The Advantages and Disadvantages of Automated Guided Vehicles (AGVs)," Conveyco, Apr. 14, 2023. [Online]. Available: <https://www.conveyco.com/blog/advantages-disadvantages-automated-guided-vehicles-agvs/> Accessed on: Mar. 14, 2024
- [34] S. User, "AGV Safety." [Online]. Available: <https://www.agvnetwork.com/automated-guided-vehicles-safety-systems> Accessed on: Jun. 20, 2024
- [35] "Scanner laser de segurança S3000 Expert - SICK." [Online]. Available: <https://www.ffonseca.com/pt/prod-scanner-laser-de-seguranca-s3000-expert-sick> Accessed on: Mar. 20, 2024
- [36] "Escáner láser SICK S30A-4011CA, 49m, 60 ms, Roscado, 185 x 155 x 160 mm, 185mm, Tamaño 3, S3000 2 | RS." [Online]. Available: <https://pt.rs-online.com/web/p/medicion-y-deteccion-laser/0271568> Accessed on: Mar. 20, 2024
- [37] "Alvium configurator," Allied Vision, Jan. 25, 2023. [Online]. Available: <https://www.alliedvision.com/en/products/alvium-configurator/alvium-1800-u/507/> Accessed on: Mar. 26, 2024
- [38] "SICK | Sensor Intelligence," SICK. [Online]. Available: <https://www.sick.com/br/pt/produtos-e-solucoes/produtos/safety/sistemas-de-camera-seguros/c/g569549> Accessed on: Mar. 26, 2024
- [39] Toposens, "Toposens Year-in-Review 2021," MEYSENS GmbH, Dec. 27, 2021. [Online]. Available: <https://toposens.com/2021/12/27/toposens-year-in-review-2021/> Accessed on: Apr. 1, 2024
- [40] "Toposens Launched ECHO ONE DK, Ultrasonic Echolocation Sensor." [Online]. Available <https://timestech.in/toposens-launched-echo-one-dk-ultrasonic-echolocation-sensor/> Accessed on: Apr. 4, 2024
- [41] R. J. Norcross, R. V. Bostelman, and J. A. Falco, "Automated Guided Vehicle Bumper Test Method Development," May 2015. doi: 10.6028/nist.ir.8029.
- [42] P. A. M. O. Duarte, "Development of Simulation Software and Calculus of the Automated Guided Vehicles Sweeps," 2014. [Online]. Available: <https://repositorio-aberto.up.pt/bitstream/10216/94314/2/31418.pdf> Accessed on: Apr. 6, 2024
- [43] "WD: modular wheel drives for AGVs | combine with motor." [Online]. Available: <https://www.nanotec.com/eu/en/products/8469-wd-modular-wheel-drives> Accessed on: Apr. 6, 2024
- [44] "luminum core pulley universal wheel 8-inch heavy-duty shock-absorbing wheel AGV robot special shock-absorbing wheel pu universal caster." [Online]. Available: <https://www.chinaglobalmall.com/products/1612116526105> Accessed on: Apr. 6, 2024

- [45] “Modular drive platform for your AGV and AGC system,” Ketterer Antriebe. [Online]. Available <https://www.ketterer-drives.com/news/products/ket-rob-modular-drive-platform-for-agv-and-agc/> Accessed on: Apr. 14, 2024
- [46] “Motor Horizontal na Tração de Rodas, Roda de Condução para AGV e empilhadeira, CE Aprovado, 24V, 48V, 72V, 96V, AC/DC - AliExpress 1420,” Aliexpress.” [Online]. Available: <https://pt.aliexpress.com/item/1005006497115643.html> Accessed on: Apr. 14, 2024
- [47] “10 Inch Mecanum Wheels on Omni-Directional AGV/AMRs-Mobile Robot Wheel Unit-HSOAR_Precision Cycloidal Reducer.” [Online]. Available: <https://en.china-hsoar.com/product-oemdz/mecanum-wheel-hsoar-wheel-reducer-precision-cycloidal-gearbox-wheel-integration.html> Accessed on: Apr. 17, 2024
- [48] E. A. D. S. Santos, "Logística baseada em AGVs," 2013. [Online]. Available: <https://repositorio-aberto.up.pt/bitstream/10216/72534/1/000157884.pdf> Accessed on: Apr. 20, 2024
- [49] AGV Battery - Wewo TM, Wewo TM (EN). [Online]. Available: <https://wewo-techmotion.com/solutions/automatic-guided-vehicle/agv-battery> Accessed on: Apr. 21, 2024
- [50] S. User, "AGV Battery Charging Systems Comparison. What's the best?" [Online]. Available: <https://www.agvnetwork.com/automated-guided-vehicles-battery-charging-solutions> Accessed on: Apr. 21, 2024
- [51] S. User, "AGV Navigation: Methods, Comparison, Pros and Cons - Illustrated Guide." [Online]. Available: <https://www.agvnetwork.com/types-of-navigation-systems-automated-guided-vehicles> Accessed on: Apr. 28, 2024
- [52] S. User, “AGV Navigation: Methods, Comparison, Pros and Cons - Illustrated Guide.” [Online]. Available: https://www.agvnetwork.com/index.php?option=com_content&view=article&id=27&catid=20 Accessed on: Apr. 28, 2024
- [53] H. Obeidat, W. Shuaieb, O. Obeidat, and R. Abd-Alhameed, "A Review of Indoor Localization Techniques and Wireless Technologies," *Wireless Personal Communications*, vol. 119, no. 1, pp. 289-327, Feb. 2021, doi: 10.1007/s11277-021-08209-5.
- [54] P. M. Rebelo, J. Lima, S. P. Soares, P. M. Oliveira, H. Sobreira, and P. Costa, "A Performance Comparison between Different Industrial Real-Time Indoor Localization Systems for Mobile Platforms," *Sensors*, vol. 24, no. 7, p. 2095, Mar. 2024, doi: 10.3390/s24072095.
- [55] HMI: Human-Machine Interface, *Inductive Automation*, Aug. 28, 2024. [Online]. Available: <https://inductiveautomation.com/resources/article/what-is-hmi> Accessed on: Apr. 28, 2024

- [56] A utilização de Interfaces HMI nas Linhas de Produção, Issuu. [Online]. Available: <https://issuu.com/ctcv/docs/revista-tecnica-setembro/s/11044427> Accessed on: May. 5, 2024
- [57] T. Suzuki et al., "A Human Interface for Interacting with and Monitoring the Multi-Agent Robotic System," in Springer eBooks, 1994, pp. 50–61. doi: 10.1007/978-4-431-68275-2_6.
- [58] Wikipedia contributors, "ENIAC," Wikipedia, Oct. 08, 2024. [Online]. Available: <https://en.wikipedia.org/wiki/ENIAC> Accessed on: May. 5, 2024
- [59] Contribuidores da Wikipédia, "Ficheiro:Used Punchcard (5151286161).jpg – Wikipédia, a enciclopédia livre." [Online]. Available: [https://pt.m.wikipedia.org/wiki/Ficheiro:Used_Punchcard_\(5151286161\).jpg](https://pt.m.wikipedia.org/wiki/Ficheiro:Used_Punchcard_(5151286161).jpg) Accessed on: May. 6, 2024
- [60] K. Cheng, "History of the Human-Machine Interface," MRO Electric Blog, Jun. 08, 2023. [Online]. Available <https://www.mroelectric.com/blog/history-of-the-human-machine-interface/> Accessed on: May. 13, 2024
- [61] P. Tróia, "Breve história do microprocessador e do computador pessoal parte 2: O fim do duopólio da Intel e da Motorola," PCGuia, Dec. 31, 2019. [Online]. Available: <https://www.pcguia.pt/2019/12/breve-historia-microprocessador-computador-pessoal-parte-2-fim-duopolio-intel-motorola/> Accessed on: May. 19, 2024
- [62] "What is a Data Logger?" [Online]. Available <https://coldroom.ae/services/data-logger/> Accessed on: May. 19, 2024
- [63] What is a Human Machine Interface (HMI)? Types, Advantages. [Online]. Available: <https://automationcommunity.com/human-machine-interface/> Accessed on: May. 27, 2024
- [64] A. Admin, "Human Machine Interface Guide," anaheimautomation, Oct. 15, 2024. [Online]. Available <https://anaheimautomation.com/blog/post/human-machine-interface-guide> Accessed on: May. 27, 2024
- [65] A. Team, "3 Types of Human-Machine Interfaces and Their Roles in Manufacturing," Global Electronic Services, Sep. 23, 2024. [Online]. Available <https://gesrepair.com/3-types-of-human-machine-interfaces-and-their-roles-in-manufacturing/> Accessed on: May. 27, 2024
- [66] "IHM com touch screen - SIMATIC," Siemens Safety Integrated - Embutida / Móvel / 800 X 480. [Online]. Available: <https://www.directindustry.com/pt/prod/siemens-safety-integrated/product-14423-2230841.html> Accessed on: Jun. 6, 2024
- [67] Discrete, "What is HMI(Human Machine Interface)? Types, Applications, Working," Discrete. [Online]. Available: <https://discrete.co.in/blog/what-is-hmi-types-applications-working> Accessed on: Jun. 6, 2024

- [68] J. P. De Wardt, "Guest Editorial: Trends in Remote Operations and Drilling Systems Automation Point to an Expanding Footprint-What Comes Next and When?," JPT, Nov. 01, 2022. [Online]. Available: <https://jpt.spe.org/trends-in-remote-operations-and-drilling-systems-automation-point-to-an-expanding-footprint-what-comes-next-and-when> Accessed on: Jun. 17, 2024
- [69] "SP5000 Series Features | Pro-face by Schneider Electric." [Online]. Available <https://www.proface.com/en-us/product/hmi/sp5000/feature> Accessed on: Jun. 17, 2024
- [70] Fast and flexible planning of new workstations in production thanks to virtual reality and digitized 3D factory data BMW Group PressClub. [Online]. Available <https://www.press.bmwgroup.com/global/article/detail/T0294345EN/absolutely-real-virtual-and-augmented-reality-open-new-avenues-in-the-bmw-group-production-system?language=en> Accessed on: Jun. 20, 2024
- [71] D. Mourtzis, J. Angelopoulos, and N. Panopoulos, "The Future of the Human-Machine Interface (HMI) in Society 5.0," Future Internet, vol. 15, no. 5, p. 162, Apr. 2023, doi: 10.3390/fi15050162.
- [72] "N. Cogotti, "AR and VR: Reshaping the Blueprint of Manufacturing Excellence," Jul. 15, 2024. [Online]. Available <https://www.linkedin.com/pulse/ar-vr-reshaping-blueprint-manufacturing-excellence-nicola-cogotti-scdrf> Accessed on: Jun. 20, 2024
- [73] J. Morgan, M. Halton, Y. Qiao, and J. G. Breslin, "Industry 4.0 smart reconfigurable manufacturing machines," Journal of Manufacturing Systems, vol. 59, pp. 481-506, Apr. 2021, doi: 10.1016/j.jmsy.2021.03.001.
- [74] Melanie, "Flexible Manufacturing Systems: Types, Examples, & Advantages," Oct. 04, 2024 [Online]. Available <https://www.unleashedsoftware.com/blog/flexible-manufacturing-in-2021-flexible-manufacturing-systems-explained/> Accessed on: Jun. 26, 2024
- [75] "Dell Servidor PowerEdge T150 Xeon E-2314 | Techinn." [Online]. Available <https://www.tradeinn.com/techinn/pt/dell-servidor-poweredge-t150-xeon-e-2314/140186741/p> Accessed on: Jun. 26, 2024
- [76] "TP-LINK Archer A5 router sem fios Dual-band (2,4 GHz / 5 GHz) Fast Ethernet Branco," Coolmática. [Online]. Available <https://www.coolmatica.pt/store/routers/76454-tp-link-archer-a5-router-sem-fios-dual-band-24-ghz-5-ghz-fast-ethernet-branco-6935364084134.html> Accessed on: Jul. 1, 2024
- [77] "AirLink - Ponto de acesso de wi-fi Wi-Fi by ACKSYS Communications & Systems | DirectIndustry." [Online]. Available <https://www.directindustry.com/pt/prod/acksys-communications-systems/product-8580-1913241.html> Accessed on: Jul. 1, 2024

- [78] “MTX-GTW (199802110) MATRIX-M2M / MTX-M2M IoT Modems / M2M IoT Modems / Wireless communication - - Maritex.” [Online]. Available https://en.maritex.com.pl/wireless_communication/m2m_iot_modems/mtx-m2m_iot_modems/m2m-mtx-gtw.html Accessed on: Jul. 6, 2024
- [79] “EUROPNEUMAQ.” [Online]. Available <https://www.europneumaq.com/pt/solucoes/robotica/robots-moveis> Accessed on: Jul. 8, 2024
- [80] “Automatic Guided Vehicles (AGVs).” [Online]. Available <https://gamrentals.com/en/news/automatic-guided-vehicles-agvs> Accessed on: Jul. 13, 2024
- [81] “EUROPNEUMAQ.” [Online]. Available <https://www.europneumaq.com/pt/solucoes/robotica/robots-moveis> Accessed on: Jul. 13, 2024
- [82] “TOOL416 2 Rolos Fita Adesiva Magnética 76×1.25cm – Radipeças.” [Online]. Available <https://www.radipecas.com/produto/tool416-2-rolos-fita-adesiva-magnetica-76x1-25cm/> Accessed on: Jul. 20, 2024
- [83] “XAMPP Installers and Downloads for Apache Friends.” [Online]. Available <https://www.apachefriends.org/> Accessed on: Jul. 20, 2024
- [84] T. Ricardo, “[cPanel] – Criar base de dados MySQL,” Alojamento Web, Jun. 18, 2011. [Online]. Available <https://blog.webtuga.pt/cpanel-criar-base-de-dados-mysql/> Accessed on: Aug. 20, 2024
- [85] LoudAdmin, “Python Programming Language: Everything You Need to Know - Loud Bench,” Loud Bench, Aug. 11, 2023. [Online]. Available <https://loudbench.com/python/> Accessed on: Aug. 3, 2024
- [86] Contribuidores da Wikipédia, “PHP,” May 01, 2024. [Online]. Available <https://pt.wikipedia.org/wiki/PHP> Accessed on: Aug. 3, 2024
- [87] N. Fepe, “JavaScript — Evolução e Especificações - Jaguaribe Tech - Medium,” Medium, Aug. 01, 2018. [Online]. Available: <https://medium.com/jaguaribetech/javascript-evolu%C3%A7%C3%A3o-e-especifica%C3%A7%C3%B5es-bb3c218e15e1> Accessed on: Aug. 17, 2024
- [88] L. Koyama, “CSS - REM e EM para nunca mais esquecer,” Jul. 27, 2024. [Online]. Available: <https://pt.linkedin.com/pulse/css-rem-e-em-para-nunca-mais-esquecer-lucas-koyama-zp9mf> Accessed on: Aug. 26, 2024
- [89] R. Terra, “Validação de formulários com HTML5,” Mar. 14, 2018. [Online]. Available: <https://pt.linkedin.com/pulse/valida%C3%A7%C3%A3o-de-formul%C3%A1rios-com-html5-rodolfo-terra> Accessed on: Set. 4, 2024
- [90] “Automatic Guided Vehicles (AGVs).” [Online]. Available: <https://gamrentals.com/en/news/automatic-guided-vehicles-agvs> Accessed on: Set. 4, 2024

ANEXOS

Anexo A – Index.php

```
1. Página inicial da plataforma web em desenvolvimento, onde é necessário realizar o login
2. <!DOCTYPE html>
3. <html lang="en">
4. <head>
5.     <meta charset="UTF-8">
6.     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7.     <title>Login</title>
8.     <link rel="stylesheet" href="https://unpkg.com/bulma@0.9.0/css/bulma.min.css" />
9.     <link rel="stylesheet" href="css/style.css?v=<?php=$version?>">
10. </head>
11. <body>
12.     <section class="area-login">
13.         <div class="login">
14.             <div>
15.                 
16.             </div>
17.             <form action="" method="post">
18.                 <input type="text" name="username" placeholder="Username">
19.                 <input type="password" name="password" placeholder="Password">
20.                 <input type="submit" name="login" value=Login>
21.             </form>
22.         </div>
23.     </section>
24. </body>
25. </html>
26. <?php
27. include('conf/conf.php');
28. /* PARA INICIAR A SESSAO E NAO CONSEGUIR ACEDER A PÁGINAS QUE NAO TEM PERMISSÃO*/
29. session_start();
30. if(isset($_SESSION['user'])){
31.     header("Location: pagina.html");
32. }
33. /* CASO O BOTÃO LOGIN SEJA PREMIDO*/
34. if(isset($_POST['login'])){
35.     $_SESSION['username'] = $_POST['username'];
36.     $_SESSION['password'] = $_POST['password'];
37.
38.     /* PARA VERIFICAR DADOS DE LOGIN! */
39.     $sql_loginusers = "SELECT *
40. FROM `users`
41. WHERE `Nome` LIKE '$_SESSION[username]'
42. AND `Idade` LIKE '$_SESSION[password]'";
43.
44.     $sql_query = mysqli_query($con, $sql_loginusers);
45.
46.     if ($sql_query->num_rows != 0) {
47.         $_SESSION['user'] = true;
48.         header("Location: pagina.html");
49.     }else{
50.         echo "Credenciais erradas!";
51.     }
52. }
53. ?>
```

Anexo B – Pagina.html

```

1. Página HOME com todas as opções
2. <!DOCTYPE html>
3. <html lang="pt-pt">
4. <head>
5.     <meta charset="UTF-8">
6.     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7.     <title>Projeto David</title>
8.     <!-- Dá importe ao ficheiro css que vai dar o estilo para toda a pagina web -->
9.     <link rel="stylesheet" href="css/style.css?v=<?php echo $version; ?>">
10.    <!-- Dá importe a uma biblioteca de JS utilizada -->
11.    <script src="https://code.jquery.com/jquery-3.6.0.min.js" defer</script>
12.    <!-- Dá import ao script de JS que foi utilizado e desenvolvido em todo o projeto -->
13.    <script src="js/script.js" defer</script>
14. </head>
15. <body>
16.     <!-- Cria um elemento para a barra de navegação que existe no topo de todas as paginas -->
17.     <div id="top-nav"></div>
18.
19.     <!-- Cria o elemento base do projeto e onde vão ser colocadas as rotas de forma visual -->
20.     <div id="imagem-container">
21.         
22.     </div>
23.
24.     <!-- Cria um botão para guardar as informações para a base de dados -->
25.     <form method="post">
26.         <button type="submit" id="GuardarTags" class="button1"
onclick="confirmarGuardarTags();">Guardar Tags</button>
27.     </form>
28.
29.     <!-- Cria um botão para depois de um selecionado vai fazer um Loop para ler a sua informação
de X em X seg -->
30.     <button type="submit" id="Loop" class="button1" onclick="Loop();">Analisar AGV</button>
31.
32.     <!-- Cria a lista de AGVs que podem selecionados para realizar a tarefa em cima descrita
-->
33.     <form id="FormAGVs" method="post">
34.         <label for="agvs">Selecione os AGVs:</label>
35.         <select name="agvs[]" id="agvs" multiple size="2">
36.             <option value="AGV.ISEC">AGV.ISEC</option>
37.             <option value="AGV.TESTE1">AGV.TESTE1</option>
38.             <option value="AGV.TESTE1">AGV.TESTE2</option>
39.         </select>
40.     </form>
41.
42.     <!-- Cria a tabela onde vão ser visualizados as tags -->
43.     <table id="tabela-pontos">
44.         <thead>
45.             <tr>
46.                 <th>TAG Number</th>
47.                 <th>X</th>
48.                 <th>Y</th>
49.                 <th class="acoes">Ações</th>
50.             </tr>
51.         </thead>
52.         <tbody>
53.         </tbody>
54.     </table>
55.
56.     <!-- Cria a tabela onde vão ser visualizados os dados do AGV selecionado -->
57.     <table id="tabela-posicao">
58.         <thead>
59.             <tr>
60.                 <th>IDAGV</th>
61.                 <th>AGV Name</th>
62.                 <th>Battery Level</th>
63.                 <th>IP</th>
64.                 <th>Route</th>
65.                 <th>Point</th>

```

```

66.         <th>Segment</th>
67.         <th>State</th>
68.         <th>Data Posição</th>
69.         <th>Número Erros</th>
70.         <th>Erros</th>
71.         <th>Data Erros</th>
72.     </tr>
73. </thead>
74. <tbody>
75.
76. </tbody>
77. </table>
78.
79. <!-- Este script esconde todo o que não interessa a primeira vez que a pág é carregada -->
80. <script>
81.     window.onload = function() {
82.         esconderTudo();
83.     };
84. </script>
85. </body>
86. </html>

```

Anexo C – Topnav.html

- Este código cria a barra de navegação no cimo da pagina web

```

2. <div class="topnav">
3.     <a class="active" href="pagina.html">Home</a>
4.     <!-- Cria um elemento com dropdown onde existem várias opções de rotas -->
5.     <div class="dropdown">
6.         <button class="dropbtn" href="pagina.html">Edição de Rotas e Tags</button>
7.         <!-- Quando escolhido uma destas rotas, o script de "Edição" e "atualizarImagem" são
executados -->
8.         <div class="dropdown-content">
9.             <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-HC-DIR')">KIT-
HC-DIR</a>
10.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-HC-ESQ')">KIT-
HC-ESQ</a>
11.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-HC-PQB')">KIT-
HC-PQB</a>
12.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-MVA')">KIT-
MVA</a>
13.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-MVM 14-19
DIR')">KIT-MVM 14-19 DIR</a>
14.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-MVM 02-
11')">KIT-MVM 02-11</a>
15.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-MVM-D 15-
19')">KIT-MVM-D 15-19</a>
16.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-MVM-G 14-
19')">KIT-MVM-G 14-19</a>
17.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-PAV')">KIT-
PAV</a>
18.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('KIT-PLC')">KIT-
PLC</a>
19.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('PAV')">PAV</a>
20.            <a href="javascript:void(0)" onclick="Edicao();atualizarImagem('PREP')">PREP</a>
21.            <a href="javascript:void(0)"
onclick="Edicao();atualizarImagem('ROTA_ISEC')">ROTA_ISEC</a>
22.        </div>
23.    </div>
24.    <div class="dropdown">
25.        <button class="dropbtn" href="pagina.html">Monitorização de AGVs</button>
26.        <!-- Quando escolhido uma destas rotas, o script de "Monitorizacao" e "atualizarImagem"
são executados -->
27.        <div class="dropdown-content">
28.            <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-HC-
DIR')">KIT-HC-DIR</a>
29.            <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-HC-
ESQ')">KIT-HC-ESQ</a>

```

```

30.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-HC-
PQB')">KIT-HC-PQB</a>
31.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-
MVA')">KIT-MVA</a>
32.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-MVM 14-
19 DIR')">KIT-MVM 14-19 DIR</a>
33.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-MVM 02-
11')">KIT-MVM 02-11</a>
34.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-MVM-D
15-19')">KIT-MVM-D 15-19</a>
35.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-MVM-G
14-19')">KIT-MVM-G 14-19</a>
36.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-
PAV')">KIT-PAV</a>
37.         <a href="javascript:void(0)" onclick="Monitorizacao();atualizarImagem('KIT-
PLC')">KIT-PLC</a>
38.                                     <a href="javascript:void(0)"
onclick="Monitorizacao();atualizarImagem('PAV')">PAV</a>
39.                                     <a href="javascript:void(0)"
onclick="Monitorizacao();atualizarImagem('PREP')">PREP</a>
40.                                     <a href="javascript:void(0)"
onclick="Monitorizacao();atualizarImagem('ROTA_ISEC')">ROTA_ISEC</a>
41.     </div>
42. </div>
43. <!-- Cria um botão onde é possível realizar o logout -->
44. <a href="logout.php">Logout</a>
45. <!-- Coloco na barra de navegação o logo da Stellantis -->
46. 
47. </div>

```

Anexo D – Logout.php

```

1. <?php
2. /* Para voltar para a página inicial quando é acionado logout! */
3. session_start();
4. session_destroy();
5. header("Location: index.php");
6. ?>

```

Anexo E – Sistema.php

```

1. //Código responsável por toda a lógica do processamento das bases de dados
2. <?php
3. // Esta parte do código dá include do script de configuração da base de dados
4. include "conf/conf.php";
5. header("Content-Type: application/json");
6.
7. // Inicia uma sessão capaz de verificar se o utilizador tem o login feito
8. session_start();
9. if (!isset($_SESSION["user"])) {
10.     header("Location: index.php");
11.     session_destroy();
12.     exit();
13. }
14.
15. // Esta função quando é executada consulta a base de dados dos AGVs (Sigat MultiAGV)
16. // Analisa os dados dos AGVs e processa-os para verificar se está tudo de acordo com o esperado
17. // Caso os dados e a sequência dos AGVs esteja correta, estes são inseridos na base de dados
do software
18. function processaAGV($NomeAGV, $conn1, &$statusMensagens) {
19.     $sql1 = "SELECT TOP (1000) [IDAGV], [AGVName], [Route], [Point] FROM
[SIGAT_EASYBOT].[dbo].[AGVInfo] WHERE [AGVName]= '$NomeAGV'";
20.     $result1 = sqlsrv_query($conn1, $sql1);
21.
22.     if ($result1 === false) {
23.         $statusMensagens[] = "Erro ao executar consulta: " . print_r(sqlsrv_errors(), true);
24.         return;
25.     } else {

```

```

26.         while ($row1 = sqlsrv_fetch_array($result1, SQLSRV_FETCH_ASSOC)) {
27.             $point = $row1['Point'];
28.             $route = $row1['Route'];
29.             $statusMensagens[] = "Ponto " . $point . " Rota " . $route;
30.
31.             // Verifica se o ponto atual é diferente do último ponto processado
32.             if (!isset($_SESSION['ponto_atual'][$NomeAGV]) || $_SESSION['ponto_atual'][$NomeAGV]
!= $point) {
33.                 $_SESSION['ponto_atual'][$NomeAGV] = $point;
34.
35.                 // Conexão com a base de dados do servidor
36.                 $conn2 = new mysqli("localhost", "root", "", "stellantis_teste");
37.                 if ($conn2->connect_error) {
38.                     $statusMensagens[] = "Conexão com MySQL falhou: " . $conn2->connect_error;
39.                     return;
40.                 }
41.
42.                 // Procura o nome da rota selecionada pelo id
43.                 $sql4 = "SELECT nome FROM rotas WHERE rota_id = '$route'";
44.                 $result4 = $conn2->query($sql4);
45.                 if ($result4 === false) {
46.                     $statusMensagens[] = "Erro na consulta MySQL: " . $conn2->error;
47.                     return;
48.                 } elseif ($result4->num_rows > 0) {
49.                     $row4 = $result4->fetch_assoc();
50.                     $Rota = $row4['nome'];
51.                     $statusMensagens[] = " Rota nome " . $Rota;
52.                 }
53.
54.                 // Procura o ponto seguinte esperado para aquela rota em específica e
55.                 // analisando o ponto atual
56.                 $sql2 = "SELECT Punto_siguiente, Tag FROM `Rota` WHERE punto = '$point'";
57.                 $result2 = $conn2->query($sql2);
58.                 if ($result2 === false) {
59.                     $statusMensagens[] = "Erro na consulta MySQL: " . $conn2->error;
60.                     return;
61.                 } elseif ($result2->num_rows > 0) {
62.                     $row2 = $result2->fetch_assoc();
63.                     $puntoSeg = $row2['Punto_siguiente'];
64.                     $tag = $row2['Tag'];
65.
66.                     if (!isset($_SESSION['ponto_esperado'][$NomeAGV]) ||
67. $_SESSION['ponto_esperado'][$NomeAGV] == $point) {
68.                         $_SESSION['ponto_esperado'][$NomeAGV] = $puntoSeg;
69.
70.                         // Se tudo tiver correto insere os dados na base de dados do software
71.                         $sql3 = "INSERT INTO pos_atual (agv_id, tag_id, time)
72. SELECT agv.agv_id, tags.tag_id, NOW()
73. FROM agv
74. JOIN tags ON agv.AGVName = '$NomeAGV' AND tags.NTag = '$tag'
75. JOIN rotas ON tags.rota_id = rotas.rota_id AND rotas.nome =
76. '$Rota'";
77.                         $result3 = $conn2->query($sql3);
78.                         $statusMensagens[] = "SQL: " . $sql3;
79.                         if ($result3 === false) {
80.                             $statusMensagens[] = "Erro ao inserir: " . $conn2->error;
81.                             return;
82.                         }
83.                         $statusMensagens[] = "Posição do " . $NomeAGV . " inserida com sucesso!
84. <br />";
85.                     } else {
86.                         $statusMensagens[] = "Sequência incorreta!, TAG " . $tag . "<br />";
87.                     }
88.                 } else {
89.                     $statusMensagens[] = "Ponto inválido ou não encontrado na tabela!<br />";
90.                 }
91.                 $conn2->close();
92.             } else {
93.                 $statusMensagens[] = "Mesmo ponto detectado.<br />";
94.             }
95.         }

```

```

93.     }
94.     sqlsrv_free_stmt($result1);
95. }
96.
97. $statusMensagens = [];
98.
99. // A parte do código que se segue é responsável por analisar a informação recebida pelo JS
pelo método POST
100. if ($_SERVER["REQUEST_METHOD"] === "POST") {
101.     $content = trim(file_get_contents("php://input"));
102.     $data = json_decode($content, true);
103.
104.     // As informação recebida é separada e organizada para posterior processamento
105.     if (
106.         isset($data["operacao"]) &&
107.         isset($data["dados"]) &&
108.         isset($data["imagem"])
109.     ) {
110.         $operacao = $data["operacao"];
111.         $dados = $data["dados"];
112.         $nomeImagem = $data["imagem"];
113.
114.         // Para que tudo funcione de forma correta é necessário assegurar que o software sabe
qual é a rota
115.         $rota_id = null;
116.         $consulta_rota = $con->prepare("SELECT rota_id FROM rotas WHERE nome = ?");
117.
118.         if ($consulta_rota) {
119.             $consulta_rota->bind_param("s", $nomeImagem);
120.             if ($consulta_rota->execute()) {
121.                 $consulta_rota->bind_result($rotaIdResultado);
122.                 if ($consulta_rota->fetch()) {
123.                     $rota_id = $rotaIdResultado;
124.                 } else {
125.                     $statusMensagens[] = "Nenhuma linha encontrada para o nome da rota: " .
$nomeImagem;
126.                 }
127.                 $consulta_rota->close();
128.             } else {
129.                 $statusMensagens[] = "Erro ao executar consulta: " . $consulta_rota->error;
130.             }
131.         } else {
132.             $statusMensagens[] = "Erro na preparação da consulta: " . $con->error;
133.         }
134.         // Caso o operador selecione uma rota, a imagem da mesma é alterada e aparecem as
tags da mesma
135.         if ($operacao === "atualizarImagem") {
136.             $consulta_tag = $con->prepare("SELECT tags.NTag, tags.X, tags.Y FROM tags WHERE
rota_id = ?");
137.             $consulta_tag->bind_param("i", $rota_id);
138.             if ($consulta_tag->execute()) {
139.                 $consulta_tag->bind_result($NTag, $X, $Y);
140.
141.                 $auxdados = [];
142.                 while ($consulta_tag->fetch()) {
143.                     $tag = [
144.                         "NTag" => $NTag,
145.                         "X" => $X,
146.                         "Y" => $Y,
147.                     ];
148.                     array_push($auxdados, $tag);
149.                 }
150.
151.                 $statusMensagens[] = json_encode($auxdados);
152.                 $consulta_tag->close();
153.             } else {
154.                 $statusMensagens[] = "Erro ao executar consulta: " . $consulta_tag->error;
155.             }
156.
157.             $data = json_decode($content, true);
158.             if ($data === null) {
159.                 $statusMensagens[] = "Erro: Dados JSON inválidos.";

```

```

160.         echo json_encode($statusMensagens);
161.         exit();
162.     }
163.
164.     // Caso o operador queira alterar a posição das tags, remover ou adicionar esta
informação é processada e armazenada
165.     }elseif ($operacao === "guardarTags") {
166.
167.         $apagarPosAtual = $con->prepare("DELETE pos_atual FROM pos_atual JOIN tags ON
pos_atual.tag_id = tags.tag_id WHERE tags.rota_id = ?");
168.         $apagarTags = $con->prepare("DELETE FROM tags WHERE rota_id = ?");
169.
170.         if ($apagarPosAtual && $apagarTags) {
171.
172.             $apagarPosAtual->bind_param("i", $rota_id);
173.             if ($apagarPosAtual->execute()) {
174.                 $apagarPosAtual->close();
175.
176.                 $apagarTags->bind_param("i", $rota_id);
177.                 if ($apagarTags->execute()){
178.                     $apagarTags->close();
179.
180.                     $inserir = $con->prepare("INSERT INTO tags (rota_id, NTag, X, Y)
VALUES (?, ?, ?, ?)");
181.
182.                     if ($inserir) {
183.                         $inserir->bind_param("iidd", $rota_id, $NTag, $X, $Y);
184.
185.                         foreach ($dados as $linha) {
186.                             $NTag = intval($linha["id"]);
187.                             $X = floatval($linha["x"]);
188.                             $Y = floatval($linha["y"]);
189.
190.                             if ($inserir->execute()) {
191.                                 $statusMensagens[] = "Inserção bem-sucedida para NTag:
$NTag";
192.                             } else {
193.                                 $statusMensagens[] = "Erro na inserção para NTag: $NTag
- " . $inserir->error;
194.                             }
195.                         }
196.                         $inserir->close();
197.                     } else {
198.                         $statusMensagens[] = "Erro na preparação da instrução SQL de
inserção - " . $con->error;
199.                     }
200.                 }else{
201.                     $statusMensagens[] = "Erro ao apagar tabela 'tags': " . $apagarPosAtual-
>error;
202.                 }
203.             } else {
204.                 $statusMensagens[] = "Erro ao apagar tabela 'tags': " . $apagarPosAtual-
>error;
205.             }
206.         } else {
207.             $statusMensagens[] = "Erro ao preparar a instrução SQL para apagar tabela
'tags'";
208.         }
209.
210.     // caso o operador queira ler a posição atual do AGV, a base de dados do software é
consultada
211. }elseif ($operacao === "lerPosAtual") {
212.     $lerPosAtual = $con->prepare("SELECT a.AGVName, a.IP, t.X, t.Y, t.NTag
213. FROM agv a
214. INNER JOIN pos_atual pa ON a.agv_id = pa.agv_id
215. INNER JOIN tags t ON pa.tag_id = t.tag_id
216. WHERE t.rota_id = ? AND pa.time = (
217.     SELECT MAX(time)
218. FROM pos_atual
219. WHERE agv_id = a.agv_id)");
220.
221.     if (!$lerPosAtual) {

```

```

222.     $statusMensagens[] = "Erro na preparação da consulta: " . $con->error;
223. } else {
224.     $lerPosAtual->bind_param("i", $rota_id);
225.     if ($lerPosAtual->execute()) {
226.         $lerPosAtual->bind_result($AGVName, $IP, $X, $Y, $NTag);
227.
228.         $auxdados123 = [];
229.         while ($lerPosAtual->fetch()) {
230.             $pos = [
231.                 "AGVName" => $AGVName,
232.                 "IP" => $IP,
233.                 "X" => $X,
234.                 "Y" => $Y,
235.                 "NTag" => $NTag
236.             ];
237.             array_push($auxdados123, $pos);
238.         }
239.
240.         $statusMensagens[] = json_encode($auxdados123);
241.         $lerPosAtual->close();
242.     } else {
243.         $statusMensagens[] = "Erro ao executar consulta: " . $lerPosAtual->error;
244.     }
245. }
246.
247. $data123 = json_decode($content, true);
248. if ($data123 === null) {
249.     $statusMensagens[] = "Erro: Dados JSON inválidos.";
250.     echo json_encode($statusMensagens);
251.     exit();
252. }
253. // Para que seja possível consultar a posição do AGV na base de dados é necessário
inserir lá a informação
254. // Esta operação é executada de X em X segundos, analisa qual é a tag onde o AGV se
encontra e chama a função "processaAGV"
255. // A função "processaAGV" consulta a base de dados dos AGV retira a informação para
a base de dados do software em desenvolvimento
256. }elseif ($operacao === "sabeTag"){
257.     // Conexão com o servidor
258.     $server = "10.220.128.53,1433";
259.     $connectionOptions = array(
260.         "Database" => "SIGAT_EASYBOT",
261.         "Uid" => "GreenAuto",
262.         "PWD" => "autogreen"
263.     );
264.     $conn1 = sqlsrv_connect($server, $connectionOptions);
265.     if ($conn1) {
266.         $statusMensagens[] ="Conexao com SQL Server estabelecida.";
267.
268.         // Processando os AGVs enviados
269.         if (isset($dados) && is_array($dados)) {
270.             foreach ($dados as $agv) {
271.                 if (isset($agv['NomeAGVs'])){
272.                     processaAGV($agv['NomeAGVs'], $conn1, $statusMensagens);
273.                     $statusMensagens[] = "Siga processar! " . $agv['NomeAGVs'];
274.
275.                 } else if (isset($agv['NAgvs'])){
276.                     $statusMensagens[] = "Numero de AGVs: " . $agv['NAgvs'];
277.
278.                 } else {
279.                     $statusMensagens[] = "Erro: Dados de AGV incompletos ou mal
formatados.";
280.                 }
281.             }
282.         } else {
283.             $statusMensagens[] = "Erro: Dados de AGVs não são um array ou estão
ausentes.";
284.         }
285.     } else {
286.         $statusMensagens[] ="Conexao com SQL Server falhou.";
287.     }
288.

```

```

289. // Esta operação pega nos dados mais relevantes da base de dados dos AGVs, realiza
uma query com eles e estes são enviados para a visualização do operador
290. }elseif ($operacao === "tabelaAGVs") {
291.     $serverName = "10.220.128.53,1433";
292.     $connectionOptions = [
293.         "Database" => "SIGAT_EASYBOT",
294.         "Uid" => "GreenAuto",
295.         "PWD" => "autogreen"
296.     ];
297.
298.     $conn = sqlsrv_connect($serverName, $connectionOptions);
299.     if (!$conn) {
300.         echo json_encode(["error" => "Conexão falhou", "details" =>
print_r(sqlsrv_errors(), true)]);
301.         exit;
302.     }
303.
304.     $data = json_decode(file_get_contents('php://input'), true);
305.     if ($_SERVER['REQUEST_METHOD'] === 'POST' || isset($data['dados'])) {
306.         $nomeAGV = $data['dados'];
307.
308.         $sqlInfo = "SELECT
309.             i.IDAGV, i.AGVName, i.BatteryLevel, i.IP, i.Route, i.Point, i.Segment,
i.State, i.LastMessageDate as DataPos,
310.             ISNULL(e.NumErros, 0) AS NumErros, ISNULL(e.Erros, '') AS Erros,
ISNULL(e.VW_EventStartDate, '') AS DataErros
311.         FROM [SIGAT_EASYBOT].[dbo].[AGVInfo] i
312.         OUTER APPLY (
313.             SELECT
314.                 COUNT(*) AS NumErros, STRING_AGG(CAST(ae.DisplayCode AS VARCHAR) + '
- ' + ae.Description, ';' ) WITHIN GROUP (ORDER BY ae.DisplayCode) AS Erros,
315.                 MAX(ae.StartDate) AS VW_EventStartDate
316.             FROM [SIGAT_EASYBOT].[dbo].[VW_AGVEvents] ae
317.             WHERE ae.AGVName = i.AGVName
318.                   AND ae.Language = 'EN'
319.                   AND ae.Finished = '0'
320.                   AND ae.Severity != 'NONE'
321.                   AND NOT ae.Description LIKE '%Erro%'
322.         ) e
323.         WHERE i.AGVName = ?";
324.
325.         $params = [$nomeAGV];
326.         $stmtInfo = sqlsrv_query($conn, $sqlInfo, $params);
327.         if ($stmtInfo === false) {
328.             echo json_encode(["error" => "Erro ao executar a consulta", "details" =>
print_r(sqlsrv_errors(), true)]);
329.             exit;
330.         }
331.
332.         $results = [];
333.         while ($row = sqlsrv_fetch_array($stmtInfo, SQLSRV_FETCH_ASSOC)) {
334.             array_push($results, $row);
335.         }
336.
337.         sqlsrv_free_stmt($stmtInfo);
338.         $statusMensagens[] = json_encode($results);
339.     } else {
340.         echo json_encode(["error" => "Erro: Dados necessários não foram enviados."]);
341.     }
342. }elseif ($operacao === "operacao") {
343.     $statusMensagens[] = "Operação desconhecida";
344. }
345. } else {
346.     $statusMensagens[] = "Parâmetros inválidos";
347. }
348.
349. echo json_encode($statusMensagens);
350. } else {
351.     echo json_encode(["error" => "Método de requisição inválido"]);
352. }
353. ?>

```

Anexo F – Script.js

```

1. Excertos do código do Script.js, a parte onde é interligado as ações do operador com as
execuções nas bases de dados, e o processamento dos dados
2.     function Loop() {
3.         var selectedAGVs = document.getElementById('agvs').selectedOptions;
4.         var InfoAGVs = [];
5.
6.         // Depois do operador selecionar os AGVs que deseja consultar essa informação é
guardada
7.         InfoAGVs.push({ NAgvs: selectedAGVs.length });
8.
9.         // São guardados os nomes de todos os AGVs selecionados
10.        for (var i = 0; i < selectedAGVs.length; i++) {
11.            InfoAGVs.push({
12.                NomeAGVs: selectedAGVs[i].value
13.            });
14.        }
15.
16.        // Caso tenha sido selecionado pelo menos um AGVs
17.        if (InfoAGVs.length > 1) {
18.            // É executado um Loop de X em X seg, capaz de ler os dados dos AGVs selecionados
e mostra-la aos operadores
19.            setInterval(function() {
20.                enviaServidor("sabeTag", InfoAGVs, imagemAtual);
21.                apagarAGVs();
22.                enviaServidor('lerPosAtual', imagemAtual, imagemAtual);
23.
24.                for (var i = 0; i < selectedAGVs.length; i++) {
25.                    enviaServidor("tabelaAGVs", selectedAGVs[i].value, imagemAtual);
26.                }
27.            }, 2000);
28.        } else {
29.            alert("Por favor, selecione pelo menos um AGV para iniciar o loop.");
30.        }
31.    }
32.
33.
34. function enviaServidor(operacao, dados, imagem) {
35.     var dadosEnvio = {
36.         operacao: operacao,
37.         dados: dados,
38.         imagem: imagem
39.     };
40.     // Esta função é a função responsável pela comunicação com o PHP, script responsável
por interagir com as bases de dados
41.     // Os dados em dadosEnvio são enviados
42.     fetch("http://localhost/Stellantis/sistema.php", {
43.         method: "POST",
44.         body: JSON.stringify(dadosEnvio),
45.         headers: {
46.             "Content-Type": "application/json; charset=utf-8"
47.         },
48.     })
49.     .then(response => response.text())
50.     .then(text => {
51.         return JSON.parse(text);
52.     })
53.     // Depois de processados pelo script em PHP, é recebida uma resposta que vai alterar
a página web de acordo com o operador necessitar
54.     .then((dadosRecebidos) => {
55.         console.log(dadosRecebidos);
56.         // Caso o operador deseje alteara a rota que pretende visualizar
57.         if(operacao == "atualizarImagem"){
58.             try {
59.                 const dados = JSON.parse(dadosRecebidos[0]);
60.                 if (Array.isArray(dados)) {
61.                     tbody.innerHTML = '';
62.                     limparPontosDaImagem();
63.
64.                     dados.forEach(objeto => {
65.                         adicionarDadosATabela(objeto.NTag, objeto.X, objeto.Y);

```

```

66.         });
67.         } else {
68.         }
69.     } catch (erro) {
70.         console.error("Erro no atualizarImagem ao processar dados recebidos:",
erro);
71.     }
72.     // Caso o operador deseje visualizar a posição do AGV selecionado na rota
selecionada
73.     }else if(operacao == "lerPosAtual"){
74.         try {
75.             const dados = JSON.parse(dadosRecebidos[0]);
76.             if (Array.isArray(dados)) {
77.                 dados.forEach((agv) => {
78.                     const { AGVName, IP, X, Y, NTag } = agv;
79.                     const cor = "red";
80.                     DesenharAGV(NTag, X, Y, AGVName, cor);
81.                 });
82.             }else {
83.             }
84.         } catch (erro) {
85.             console.error("Erro no lerPosAtual ao processar dados recebidos:", erro);
86.         }
87.         // É executado sempre antes de ser mostrado qual a posição do AGV para atualizar
a base de dados da plataforma web
88.     }else if(operacao == "sabeTag"){
89.         try {
90.             console.log("Dados recebidos:", dadosRecebidos[0]);
91.         } catch (erro) {
92.             console.error("Erro no sabeTag ao processar dados recebidos:", erro);
93.         }
94.         // Recolhe a informação da base de dados dos AGV (Sigat MultiAGV) e mostra ao
operador com a utilização de uma tabela
95.     }else if(operacao == "tabelaAGVs"){
96.         try {
97.             console.log("Dados recebidos:", dadosRecebidos);
98.             //console.log("Tipo dos dados recebidos:", typeof dadosRecebidos);
99.             const tbody = document.querySelector('#tabela-posicao tbody');
100.            tbody.innerHTML = '';
101.
102.            if (Array.isArray(dadosRecebidos)) {
103.                dadosRecebidos.forEach(agv => {
104.                    adicionarDadosATabelaAGV(agv, tbody);
105.                });
106.            } else {
107.                console.error("Erro: dados recebidos não são um array", dadosRecebidos);
108.            }
109.
110.        } catch (erro) {
111.            console.error("Erro no sabeTag ao processar dados recebidos:", erro);
112.        }
113.    }else{
114.        console.log("Nenhuma operação!");
115.    }
116. })
117. .catch((error) => {
118.     console.error("Erro na requisição fetch:", error);
119. });
120. }

```

Anexo G – DB_Projeto_Tese.sql

```

1. -- Base de dados: `DB_Projeto_Tese`
2. --
3.
4. -- -----
5.
6. --
7. -- Estrutura da tabela `agv`
8. --

```

```

9.
10. CREATE TABLE `agv` (
11.   `agv_id` int(20) NOT NULL,
12.   `AGVName` varchar(200) NOT NULL,
13.   `IP` varchar(200) NOT NULL
14. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
15.
16. -----
17.
18. --
19. -- Estrutura da tabela `coiffage`
20. --
21.
22. CREATE TABLE `coiffage` (
23.   `punto_id` int(11) NOT NULL,
24.   `Ruta` int(11) NOT NULL,
25.   `Punto` int(11) NOT NULL,
26.   `Tag` int(11) NOT NULL,
27.   `Punto_siguiente` int(11) NOT NULL
28. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
29.
30. -----
31.
32. --
33. -- Estrutura da tabela `hc_dir`
34. --
35.
36. CREATE TABLE `hc_dir` (
37.   `punto_id` int(11) NOT NULL,
38.   `Ruta` int(11) NOT NULL,
39.   `Punto` int(11) NOT NULL,
40.   `Tag` int(11) NOT NULL,
41.   `Punto_siguiente` int(11) NOT NULL
42. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
43.
44. -----
45.
46. --
47. -- Estrutura da tabela `hc_esq`
48. --
49.
50. CREATE TABLE `hc_esq` (
51.   `punto_id` int(11) NOT NULL,
52.   `Ruta` int(11) NOT NULL,
53.   `Punto` int(11) NOT NULL,
54.   `Tag` int(11) NOT NULL,
55.   `Punto_siguiente` int(11) NOT NULL
56. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
57.
58. -----
59.
60. --
61. -- Estrutura da tabela `mva`
62. --
63.
64. CREATE TABLE `mva` (
65.   `punto_id` int(11) NOT NULL,
66.   `Ruta` int(11) NOT NULL,
67.   `Punto` int(11) NOT NULL,
68.   `Tag` int(11) NOT NULL,
69.   `Punto_siguiente` int(11) NOT NULL
70. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
71.
72. -----
73.
74. --
75. -- Estrutura da tabela `mvm_2-11`
76. --
77.
78. CREATE TABLE `mvm_2-11` (
79.   `punto_id` int(11) NOT NULL,
80.   `Ruta` int(11) NOT NULL,

```

```
81. `Punto` int(11) NOT NULL,
82. `Tag` int(11) NOT NULL,
83. `Punto_siguiente` int(11) NOT NULL
84. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
85.
86. -----
87.
88. --
89. -- Estrutura da tabela `mvm_14-19_dir`
90. --
91.
92. CREATE TABLE `mvm_14-19_dir` (
93. `punto_id` int(11) NOT NULL,
94. `Ruta` int(11) NOT NULL,
95. `Punto` int(11) NOT NULL,
96. `Tag` int(11) NOT NULL,
97. `Punto_siguiente` int(11) NOT NULL
98. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
99.
100. -----
101.
102. --
103. -- Estrutura da tabela `pav`
104. --
105.
106. CREATE TABLE `pav` (
107. `punto_id` int(11) NOT NULL,
108. `Ruta` int(11) NOT NULL,
109. `Punto` int(11) NOT NULL,
110. `Tag` int(11) NOT NULL,
111. `Punto_siguiente` int(11) NOT NULL
112. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
113.
114. -----
115.
116. --
117. -- Estrutura da tabela `plc`
118. --
119.
120. CREATE TABLE `plc` (
121. `punto_id` int(11) NOT NULL,
122. `Ruta` int(11) NOT NULL,
123. `Punto` int(11) NOT NULL,
124. `Tag` int(11) NOT NULL,
125. `Punto_siguiente` int(11) NOT NULL
126. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
127.
128. -----
129.
130. --
131. -- Estrutura da tabela `pos_atual`
132. --
133.
134. CREATE TABLE `pos_atual` (
135. `pos_atual_id` int(20) NOT NULL,
136. `agv_id` int(11) NOT NULL,
137. `tag_id` int(11) NOT NULL,
138. `time` datetime NOT NULL
139. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
140.
141. -----
142.
143. --
144. -- Estrutura da tabela `pqb`
145. --
146.
147. CREATE TABLE `pqb` (
148. `punto_id` int(11) NOT NULL,
149. `Ruta` int(11) NOT NULL,
150. `Punto` int(11) NOT NULL,
151. `Tag` int(11) NOT NULL,
152. `Punto_siguiente` int(11) NOT NULL
```

```

153. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
154.
155. -- -----
156.
157. --
158. -- Estrutura da tabela `prep._porteurs`
159. --
160.
161. CREATE TABLE `prep._porteurs` (
162.   `punto_id` int(11) NOT NULL,
163.   `Ruta` int(11) NOT NULL,
164.   `Punto` int(11) NOT NULL,
165.   `Tag` int(11) NOT NULL,
166.   `Punto_siguiente` int(11) NOT NULL
167. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
168.
169. -- -----
170.
171. --
172. -- Estrutura da tabela `rotas`
173. --
174.
175. CREATE TABLE `rotas` (
176.   `rota_id` int(11) NOT NULL,
177.   `nome` varchar(255) DEFAULT NULL
178. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
179.
180. -- -----
181.
182. --
183. -- Estrutura da tabela `rota_isec`
184. --
185.
186. CREATE TABLE `rota_isec` (
187.   `punto_id` int(11) NOT NULL,
188.   `Ruta` int(11) NOT NULL,
189.   `Punto` int(11) NOT NULL,
190.   `Tag` int(11) NOT NULL,
191.   `Punto_siguiente` int(11) NOT NULL
192. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
193.
194. -- -----
195.
196. --
197. -- Estrutura da tabela `tags`
198. --
199.
200. CREATE TABLE `tags` (
201.   `tag_id` int(11) NOT NULL,
202.   `rota_id` int(11) DEFAULT NULL,
203.   `NTag` int(11) DEFAULT NULL,
204.   `X` float DEFAULT NULL,
205.   `Y` float DEFAULT NULL
206. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
207.
208. -- -----
209.
210. --
211. -- Estrutura da tabela `users`
212. --
213.
214. CREATE TABLE `users` (
215.   `user_id` int(20) NOT NULL,
216.   `Nome` varchar(200) DEFAULT NULL,
217.   `Password` int(20) DEFAULT NULL,
218.   `Email` varchar(300) DEFAULT NULL
219. ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
220.
221. -- -----
222.
223. --
224. -- Índices para tabelas despejadas

```

```
225. --
226.
227. --
228. -- Índices para tabela `agv`
229. --
230. ALTER TABLE `agv`
231.   ADD PRIMARY KEY (`agv_id`);
232.
233. --
234. -- Índices para tabela `coiffage`
235. --
236. ALTER TABLE `coiffage`
237.   ADD PRIMARY KEY (`punto_id`);
238.
239. --
240. -- Índices para tabela `hc_dir`
241. --
242. ALTER TABLE `hc_dir`
243.   ADD PRIMARY KEY (`punto_id`);
244.
245. --
246. -- Índices para tabela `hc_esq`
247. --
248. ALTER TABLE `hc_esq`
249.   ADD PRIMARY KEY (`punto_id`);
250.
251. --
252. -- Índices para tabela `mva`
253. --
254. ALTER TABLE `mva`
255.   ADD PRIMARY KEY (`punto_id`);
256.
257. --
258. -- Índices para tabela `mvm_2-11`
259. --
260. ALTER TABLE `mvm_2-11`
261.   ADD PRIMARY KEY (`punto_id`);
262.
263. --
264. -- Índices para tabela `mvm_14-19_dir`
265. --
266. ALTER TABLE `mvm_14-19_dir`
267.   ADD PRIMARY KEY (`punto_id`);
268.
269. --
270. -- Índices para tabela `pav`
271. --
272. ALTER TABLE `pav`
273.   ADD PRIMARY KEY (`punto_id`);
274.
275. --
276. -- Índices para tabela `plc`
277. --
278. ALTER TABLE `plc`
279.   ADD PRIMARY KEY (`punto_id`);
280.
281. --
282. -- Índices para tabela `pos_atual`
283. --
284. ALTER TABLE `pos_atual`
285.   ADD PRIMARY KEY (`pos_atual_id`),
286.   ADD KEY `agv_id` (`agv_id`),
287.   ADD KEY `tag_id` (`tag_id`);
288.
289. --
290. -- Índices para tabela `pqb`
291. --
292. ALTER TABLE `pqb`
293.   ADD PRIMARY KEY (`punto_id`);
294.
295. --
296. -- Índices para tabela `prep._porteurs`
```

```
297. --
298. ALTER TABLE `prep._porteurs`
299.   ADD PRIMARY KEY (`punto_id`);
300.
301. --
302. -- Índices para tabela `rotas`
303. --
304. ALTER TABLE `rotas`
305.   ADD PRIMARY KEY (`rota_id`);
306.
307. --
308. -- Índices para tabela `rota_isec`
309. --
310. ALTER TABLE `rota_isec`
311.   ADD PRIMARY KEY (`punto_id`);
312.
313. --
314. -- Índices para tabela `tags`
315. --
316. ALTER TABLE `tags`
317.   ADD PRIMARY KEY (`tag_id`),
318.   ADD KEY `rota_id` (`rota_id`);
319.
320. --
321. -- Limitadores para a tabela `pos_atual`
322. --
323. ALTER TABLE `pos_atual`
324.   ADD CONSTRAINT `fk_pos_atual_agv` FOREIGN KEY (`agv_id`) REFERENCES `agv` (`agv_id`),
325.   ADD CONSTRAINT `fk_pos_atual_tags` FOREIGN KEY (`tag_id`) REFERENCES `tags` (`tag_id`);
326.
327. --
328. -- Limitadores para a tabela `tags`
329. --
330. ALTER TABLE `tags`
331.   ADD CONSTRAINT `fk_tags_rotas` FOREIGN KEY (`rota_id`) REFERENCES `rotas` (`rota_id`);
332. COMMIT;
```