

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Desenvolvimento de um Sistema de Controlo Inteligente para o AirLight para Integração em Smart Buildings

João Pedro Morais da Silva



Mestrado em Engenharia Eletrotécnica e de Computadores

Orientador: José Rui da Rocha Pinto Ferreira

Supervisor: João Brandão

30 de julho de 2026

Resumo

A iluminação e a ventilação natural constituem estratégias passivas com impacto direto na eficiência energética e no conforto dos ocupantes de edifícios. Em edifícios industriais e terciários, estas estratégias são frequentemente proporcionadas por sistemas de abertura instalados em cobertura, cuja posição de abertura determina o desempenho luminoso e o caudal de ventilação do espaço. A exploração eficaz destes sistemas depende de decisões concretas, que envolvem quando e como é que o sistema deve abrir, dado que, sem uma lógica de controlo adequada, estes ficam dependentes da intervenção manual dos ocupantes. O *AirLight* é um equipamento deste tipo, integrando iluminação natural difusa e ventilação natural num único sistema, mas que, à data de início deste trabalho, não dispunha de mecanismos de controlo para responder a estas questões de forma automatizada.

A presente dissertação, desenvolvida em colaboração com a empresa Chatron, descreve o desenvolvimento de um sistema de controlo para o equipamento em questão. A lógica implementada é responsável por gerir os modos de operação do sistema e assegurar o posicionamento em aberturas intermédias, respondendo a comandos do utilizador, a sensores ambientais e a regras de supervisão. O trabalho aborda também a forma como sistemas externos podem interagir com a lógica de controlo desenvolvida, tendo sido estudada e implementada uma arquitetura de comunicação que permite a consulta do estado do sistema e o envio de comandos. Sobre esta base foram desenvolvidas interfaces de utilizador e uma camada de supervisão experimental que implementa estratégias de gestão orientadas ao conforto térmico, à qualidade do ar, à eficiência energética e à proteção do espaço. O trabalho aborda ainda a integração com sistemas de segurança contra incêndio e a viabilidade de comunicação através de protocolos industriais normalizados.

A validação do sistema foi realizada através de ensaios experimentais e de verificação funcional. Os ensaios incidiram sobre o desempenho da arquitetura de comunicação implementada e sobre o comportamento do equipamento ao longo do tempo. A verificação funcional confirmou o cumprimento dos requisitos definidos para a lógica de controlo, a interação com o utilizador e a integração com os restantes componentes do sistema. Os resultados obtidos demonstram que o sistema desenvolvido opera de acordo com o pretendido e é aplicável ao contexto operacional para o qual foi concebido.

Palavras-chave: AirLight, Ventilação, Iluminação, Controlo, Supervisão

Abstract

Natural lighting and ventilation are passive strategies with a direct impact on energy efficiency and occupant comfort in buildings. In industrial and commercial buildings, these strategies are often provided by roof-mounted opening systems, whose opening position determines both the daylighting performance and the ventilation airflow of the space. The effective use of such systems depends on concrete decisions involving when and how the system should open, since, without an adequate control logic, they remain dependent on manual intervention by the occupants. The AirLight is one such system, combining diffuse natural lighting and natural ventilation in a single unit, but which, at the start of this work, lacked the mechanisms of control to address these questions in an automated manner.

This dissertation, developed in collaboration with the company Chatron, describes the development of a control system for the equipment in question. The implemented logic is responsible for managing the operating modes of the system and ensuring positioning at intermediate openings, responding to user commands, environmental sensors and supervisory rules. The work also addresses how external systems can interact with the developed control logic, with a communication architecture having been studied and implemented to allow querying the system state and issuing commands. On this basis, user interfaces and an experimental supervisory layer were developed, implementing management strategies oriented towards thermal comfort, air quality, energy efficiency and space protection. The work further addresses the integration with fire safety systems and the feasibility of communication through standardised industrial protocols.

System validation was carried out through experimental tests and functional verification. The tests focused on the performance of the implemented communication architecture and on equipment behaviour over time. Functional verification confirmed the fulfilment of the defined requirements for control logic, user interaction and integration with the remaining system components. The results demonstrate that the developed system operates as intended and is applicable to the operational context for which it was designed.

Keywords: AirLight, Ventilation, Lighting, Control, Supervision

Objetivos de Desenvolvimento Sustentável (ODS) das Nações Unidas

Os Objetivos de Desenvolvimento Sustentável (ODS) da Organização das Nações Unidas (ONU) constituem um conjunto de 17 metas globais estabelecidas em 2015, com o propósito de promover o desenvolvimento sustentável à escala mundial até 2030. Estes objetivos abordam desafios globais como a eficiência energética, a sustentabilidade ambiental, a qualidade de vida nos centros urbanos e a utilização responsável de recursos naturais, como ilustra a Figura 1.



Figura 1: Objetivos de Desenvolvimento Sustentável da ONU.

O presente trabalho insere-se no domínio da automação de edifícios e das estratégias passivas de conforto, tendo como principal objetivo o desenvolvimento de um sistema de controlo para um equipamento que integra iluminação e ventilação natural. A adoção de soluções passivas controladas de forma automatizada contribui para a redução da dependência de sistemas ativos de climatização e iluminação artificial, promovendo uma utilização mais racional dos recursos energéticos nos edifícios. O sistema de supervisão implementado reforça esta vertente ao introduzir estratégias orientadas à eficiência energética, à qualidade do ar interior e à coordenação entre sistemas ativos e passivos.

Neste contexto, identificam-se três ODS para os quais o trabalho desenvolvido apresenta contribuições relevantes:

- **ODS 7** — Garantir o acesso a energia acessível, fiável, sustentável e moderna para todos.

- **ODS 11** — Tornar as cidades e os aglomerados urbanos inclusivos, seguros, resilientes e sustentáveis.
- **ODS 12** — Assegurar padrões de consumo e produção sustentáveis.

A Tabela 1 apresenta uma análise dos (ODS) que se revelam mais pertinentes no contexto do presente projeto, onde para cada objetivo identificado, é indicada a meta específica com a qual o projeto se relaciona, bem como a respetiva contribuição do mesmo.

Tabela 1: Contribuição do trabalho para os Objetivos de Desenvolvimento Sustentável.

ODS	Meta	Contribuição
7	7.3	O sistema de controlo desenvolvido promove a eficiência energética nos edifícios ao automatizar a gestão da ventilação e iluminação natural, reduzindo o recurso a sistemas mecânicos de climatização e a iluminação artificial.
11	11.6	A integração de estratégias de supervisão que coordenam ventilação natural, qualidade do ar e conforto térmico contribui para a redução do impacto ambiental dos edifícios urbanos, promovendo ambientes interiores mais saudáveis e sustentáveis.
12	12.2	A deteção de ineficiências implementada no sistema de supervisão, como a operação simultânea de climatização ativa com ventilação natural aberta, apoia a gestão sustentável dos recursos energéticos ao identificar e prevenir desperdícios.

Agradecimentos

Em primeiro lugar, quero agradecer à minha família, pelo apoio incondicional ao longo de todo o meu percurso académico.

Aos meus colegas e amigos, agradeço a companhia e a partilha ao longo destes anos. Em particular, aos meus colegas de curso: Vasco, Maria e Matilde, por todos os momentos que passámos juntos e por me motivarem todos os dias ao longo destes anos.

À Camila, agradeço por ser o meu porto seguro em todos os momentos e pela leveza que trouxe à minha vida.

Ao Professor José Rui Ferreira, agradeço a disponibilidade e o acompanhamento prestados durante todas as fases deste trabalho, bem como as sugestões que contribuíram diretamente para a qualidade final desta dissertação.

À empresa CHATRON agradeço pela forma como me acolheram ao longo deste semestre e por terem proporcionado as condições necessárias para o trabalho desenvolvido. Ao Eng. João Brandão agradeço pela orientação prestada ao longo de todo o trabalho, desde a fase preparatória até à sua conclusão, bem como o acompanhamento próximo e o contributo decisivo na construção e direção desta dissertação. Ao Eng. Paulo, agradeço o apoio nas questões específicas de desenvolvimento técnico, tendo sido fundamental para a obtenção dos resultados finais desta dissertação, e também para o esclarecimento de dúvidas, o que me fez aprender muito.

Declaração de Honra

Declaro que a presente dissertação é de minha autoria e não foi previamente apresentada e avaliada nesta ou em outra instituição. As referências a outros autores (afirmações, ideias, pensamentos), assim como a utilização de trabalhos publicados respeitam escrupulosamente as regras da atribuição, e encontram-se devidamente indicadas no texto e nas referências bibliográficas, de acordo com as normas de referência. Tenho consciência de que a prática de plágio ou de autoplágio constitui ilícitos académicos, conforme estabelecido no Código de Ética da U.Porto.

Declaro, ainda, que utilizei ferramentas de Inteligência Artificial para a realização de parte(s) da presente dissertação, e essa utilização e os seus resultados estão devidamente assinalados e foram objeto de cuidada verificação para deteção de erros, imprecisões, atribuições infundadas ou outras formas de enviesamento de conteúdos e argumentos, bem como de determinação de autoria.

João Pedro Morais da Silva

João Pedro Morais da Silva

"We are what we repeatedly do. Excellence, then, is not an act, but a habit."

Will Durant

Índice

1	Introdução	1
1.1	Enquadramento e Motivação	1
1.2	Contexto da Empresa e do <i>AirLight</i>	2
1.3	Objetivos do Trabalho	2
1.4	Estrutura do Documento	3
2	Enquadramento Técnico	4
2.1	Iluminação e Ventilação Natural em Edifícios	4
2.1.1	Quadro normativo de conforto térmico e qualidade do ar	5
2.1.2	Estratégias intrínsecas de controlo	6
2.1.3	Necessidade de coordenação automática	7
2.2	Sistemas de Gestão de Edifícios	7
2.2.1	Estratégias supervisoras	8
2.2.2	Quadro normativo automação predial	9
2.2.3	Limitações e relação com controladores locais	10
2.3	Autómatos Programáveis e Protocolos de Comunicação	10
2.3.1	<i>Modbus</i>	11
2.3.2	REST sobre HTTP	13
2.3.3	Critérios de escolha e contextos de aplicação	14
2.4	O Sistema <i>AirLight</i>	15
2.5	Sistemas de Desenfumagem	17
2.5.1	Análise de Soluções Comerciais	17
2.5.2	Padrões Arquiteturais	19
2.6	Síntese do Enquadramento	19
3	Sistema e Componentes	21
3.1	Visão Geral do Sistema	21
3.2	Atuadores e Fins de Curso	22
3.3	Controlo Local	23
3.4	Alimentação, Sensores e Entrada	26
3.5	Estágio de comutação	30
3.6	Esquema de Ligações	31
3.7	Estudo de integração com uma central de desenfumagem	32
3.8	Síntese do sistema	37
4	Desenvolvimento e Integração do Sistema	38
4.1	Arquitetura da lógica de controlo	38
4.2	Controlo de Posição	41

4.2.1	Método utilizado para o controlo de posição	41
4.2.2	Funcionalidade de movimento por percentagem	42
4.2.3	Inicialização do sistema	44
4.2.4	Calibração da estimativa	45
4.3	Servidor HTTP e interface de comunicação	46
4.3.1	Variáveis partilhadas entre <i>sketch</i> e PLC	46
4.3.2	<i>Endpoints</i> REST	48
4.4	Interface web	49
4.4.1	Enquadramento e tecnologia da interface web	50
4.4.2	Compatibilidade com o servidor desenvolvido	50
4.5	Padrões de comunicação	51
4.6	Aplicação móvel	53
4.6.1	Funcionalidades Principais	53
4.6.2	Estatísticas e Modais de Aviso	56
4.6.3	Compilação e distribuição da aplicação	58
4.7	Integração com sistemas através de Modbus	58
4.8	Implementação experimental de BMS com Home Assistant	61
4.8.1	Enquadramento e escolha da plataforma	61
4.8.2	Arquitetura de integração	62
4.8.3	Modelação das entradas	62
4.8.4	Estratégias de supervisão implementadas	63
4.9	Estudo de Integração com Central de Desenfumagem	66
4.10	Síntese do Desenvolvimento	68
5	Resultados Experimentais	69
5.1	Caracterização do desempenho de comunicação e resposta	69
5.1.1	Latência de Comunicação	69
5.1.2	Tempo Total de Resposta	71
5.1.3	Discussão de resultados	72
5.2	Caracterização dos tempos de curso do atuador	72
6	Conclusões e Trabalho Futuro	74
6.1	Conclusões	74
6.2	Trabalho Futuro	75
6.2.1	Consolidação do sistema	75
6.2.2	Integração industrial	76
A	Código AppOpta	80

Lista de Figuras

1	Objetivos de Desenvolvimento Sustentável da ONU.	iii
2.1	Exemplo de arquitetura de rede <i>Modbus</i> . Adaptado de [24].	11
2.2	Sistema <i>AirLight</i>	15
2.3	<i>AirLight</i> na posição aberta.	16
3.1	Diagrama de blocos geral do sistema.	21
3.2	Atuador eletromecânico.	22
3.3	Fins de curso instalados.	23
3.4	Arduino Opta.	24
3.5	<i>Pinout</i> Opta.	25
3.6	Fonte de alimentação.	27
3.7	Botoneiras para controlo manual.	28
3.8	Sensor de temperatura.	29
3.9	Sensor de chuva.	29
3.10	Relés do estágio de comutação.	30
3.11	Esquema de funcionamento dos Relés.	31
3.12	Esquemático das ligações elétricas do sistema de controlo.	31
3.13	Disposição dos componentes em calha DIN e sensores.	32
3.14	Central de Desenfumagem Actulux.	33
3.15	Dispositivo UniMon.	34
3.16	Painel de Incêndio.	35
3.17	Configuração da Central de Desenfumagem.	36
3.18	Esquemático das ligações elétricas na configuração estudada com central de de- senfumagem.	37
4.1	Máquina de estados <i>ModeMachine</i>	39
4.2	Programa <i>RelayControl</i>	40
4.3	Máquina de estados <i>AutoControl</i>	41
4.4	Configuração Wi-Fi no Arduino PLC IDE.	47
4.5	<i>Shared Variables</i>	48
4.6	Interface web desenvolvida	50
4.7	Ecrã de inicialização da aplicação móvel.	54
4.8	Secções principais de controlo na aplicação móvel.	55
4.9	Secção de Estatísticas da aplicação móvel.	56
4.10	Modais relativos ao sensor de temperatura.	57
4.11	Configuração de parâmetros Ethernet no Arduino PLC IDE.	59
4.12	Simulador qModMaster.	60
4.13	Declaração de variáveis como <i>Parameters</i>	60

4.14	Interface Home Assistant.	63
4.15	Estratégias de supervisão e estado global do edifício.	65
4.16	Indicadores de consumo e detecção de ineficiências.	65
5.1	Histograma da latência de comunicação do comando de alteração de modo, por rede.	70
5.2	Comparação da latência de comunicação do comando de alteração de modo entre redes (<i>boxplot</i>).	71
5.3	Evolução temporal da latência de comunicação ao longo das amostras recolhidas, por rede.	71
5.4	Comparação do tempo total de resposta ao comando de posicionamento entre redes, por sentido de movimento (<i>boxplot</i>).	72

Lista de Tabelas

1	Contribuição do trabalho para os Objetivos de Desenvolvimento Sustentável. . .	iv
2.1	Categorias de qualidade do ar interior e conforto térmico segundo a norma EN 16798-1.	6
2.2	Fatores de eficiência energética segundo a norma EN 15232.	10
2.3	Funções Modbus.	12
3.1	Versões do Arduino Opta.	25
5.1	Latência de comunicação entre cliente e OPTA, por rede.	70
5.2	Tempos de abertura e fecho registados nos ensaios de caracterização.	73

Lista de Abreviaturas

API	Application Programming Interface
BMS	Building Management Systems
SCADA	Supervisory Control and Data Acquisition
CORS	Cross-Origin Resource Sharing
HTTP	Hypertext Transfer Protocol
HVAC	Heating, Ventilation and Air Conditioning
IEC	International Electrotechnical Commission
IoT	Internet of Things
JSON	JavaScript Object Notation
PLC	Programmable Logic Controller
REST	Representational State Transfer
SHEV	Smoke and Heat Exhaust Ventilation systems
ST	Structured Text
LD	Ladder Diagram
DC	Direct Current
TCP	Transmission Control Protocol

Capítulo 1

Introdução

1.1 Enquadramento e Motivação

A exigência crescente de eficiência energética e de qualidade do ambiente interior nos edifícios tem levado à adoção de soluções passivas de iluminação e ventilação natural. Em edifícios industriais, onde a iluminação artificial e a climatização mecânica representam uma parcela significativa do consumo energético, estas estratégias podem contribuir de forma relevante para a redução de custos operacionais e para a melhoria do conforto dos ocupantes. Estas estratégias são frequentemente proporcionadas por sistemas de abertura instalados em cobertura, cuja posição de abertura determina o desempenho luminoso e o caudal de ventilação do espaço.

No entanto, as soluções convencionais, como as claraboias tradicionais, apresentam uma limitação frequente: a sua abertura para efeitos de ventilação introduz radiação solar direta no espaço interior, o que pode causar desconforto nos ocupantes, criar marcas de luz intensa no pavimento e inviabilizar postos de trabalho situados sob a cobertura. Este conflito entre a necessidade de ventilar e a necessidade de controlar a incidência luminosa leva, em muitos casos, a que estes sistemas permaneçam fechados, anulando o seu potencial de ventilação natural.

A eficácia destas soluções depende, contudo, não apenas da conceção física do equipamento, mas também da forma como este é gerido. Sistemas operados manualmente ou com lógicas simplificadas apresentam desempenho variável, dependente do comportamento dos utilizadores e das condições exteriores. A aplicação de estratégias de controlo automatizado, capazes de reagir a sensores e a previsões ambientais, permite uma exploração mais consistente e eficiente destes sistemas. Contudo, muitos dos equipamentos disponíveis no mercado para iluminação e ventilação em cobertura continuam a operar com lógicas de comando simplificadas ou dependentes de intervenção manual, sem uma camada de controlo que permita explorar o seu potencial de forma integrada e autónoma. É neste contexto que se insere o presente trabalho.

1.2 Contexto da Empresa e do *AirLight*

A empresa Chatron desenvolve soluções técnicas para equipamentos de iluminação natural, ventilação e desenfumagem, aplicados maioritariamente em edifícios do setor industrial e terciário. Neste contexto, surge o *AirLight*, um equipamento concebido para responder ao conflito descrito anteriormente, integrando iluminação natural difusa e ventilação natural num único sistema aplicado em cobertura. Este é constituído por um difusor, responsável por assegurar uma luz uniformemente distribuída, eliminando a incidência solar direta que caracteriza as claraboias convencionais. A ventilação é proporcionada por uma abertura elevatória, acionada por um atuador elétrico, em que o fluxo de ar é direcionado pelas laterais do equipamento, garantindo ventilação eficaz sem exposição direta da abertura ao exterior, o que protege o espaço contra a entrada de elementos indesejados.

Apesar de o *AirLight* resolver o problema físico da coexistência entre ventilação e iluminação natural, à data de início deste trabalho o equipamento dispunha apenas de um sistema de comando local baseado em botões físicos ou interruptores. Nesta configuração, não era possível efetuar o seu controlo remoto ou integrá-lo em sistemas de supervisão e gestão técnica de edifícios. Paralelamente, inexistia uma lógica de controlo estruturada que permitisse gerir diferentes modos de funcionamento segundo uma hierarquia de prioridades, bem como uma interface de comunicação normalizada que possibilitasse a interação com aplicações externas. Neste contexto, o desafio proposto pela empresa e o objetivo principal desta dissertação consistiram no desenvolvimento de um sistema de controlo inteligente para o *AirLight*, capaz de responder autonomamente às questões fundamentais da sua operação — quando abrir, quanto abrir e em função de que condições —, dotando simultaneamente o equipamento de capacidades de comunicação, supervisão e integração compatíveis com os atuais sistemas de automação de edifícios.

1.3 Objetivos do Trabalho

O presente trabalho tem como objetivo principal o desenvolvimento de um sistema de controlo para o *AirLight* que permita a operação automatizada do equipamento e a interação remota com o utilizador. Em particular, pretende-se:

- Analisar o estado da arte em estratégias de ventilação e iluminação natural, protocolos de comunicação em automação de edifícios e soluções comerciais comparáveis;
- Desenvolver uma lógica de controlo estruturada, com suporte a múltiplos modos de operação e gestão de sensores ambientais;
- Implementar uma arquitetura de comunicação que permita a interação com o sistema a partir de aplicações externas;
- Desenvolver interfaces de utilizador para controlo e monitorização do sistema;

- Estudar a integração do equipamento com sistemas de supervisão e com outros equipamentos do edifício;
- Validar experimentalmente o desempenho do sistema desenvolvido.

1.4 Estrutura do Documento

O Capítulo 2 apresenta o enquadramento teórico e o estado da arte nos domínios relevantes para o projeto. O Capítulo 3 descreve o sistema desenvolvido ao nível dos seus componentes físicos e da arquitetura de *hardware* adotada. O Capítulo 4 detalha o desenvolvimento e a integração do sistema, abrangendo a lógica de controlo, as interfaces de comunicação e a camada de supervisão. O Capítulo 5 apresenta os resultados experimentais obtidos e o Capítulo 6 expõe as conclusões e identifica as vias de trabalho futuro.

Capítulo 2

Enquadramento Técnico

O presente capítulo estabelece o enquadramento teórico e técnico que suporta o desenvolvimento do sistema de controlo inteligente para o *AirLight*. A revisão da literatura centra-se nos domínios relevantes para o projeto, procurando evidenciar os conceitos envolvidos e também as limitações das abordagens existentes que motivaram o trabalho desenvolvido. A análise parte dos princípios da iluminação e ventilação natural em edifícios, demonstrando que o seu desempenho efetivo depende da existência de estratégias de controlo automático. São depois abordadas as tecnologias que tornam esse controlo possível, culminando numa análise comparativa de soluções comerciais que permite identificar as lacunas que o presente trabalho procura resolver.

2.1 Iluminação e Ventilação Natural em Edifícios

A iluminação e a ventilação naturais constituem estratégias passivas centrais no projeto de edifícios modernos, com impacto direto na eficiência energética e na qualidade do ambiente interior. Em edifícios industriais e terciários de grande volume, a sua relevância é particularmente acentuada, uma vez que a iluminação artificial e os sistemas de climatização mecânica representam uma parcela significativa do consumo energético global [1].

A disponibilidade de luz natural durante o período diurno permite reduzir substancialmente a necessidade de iluminação artificial. Para além da componente energética, a exposição a padrões de iluminação natural tem efeitos positivos no ritmo circadiano, no humor e no desempenho cognitivo dos ocupantes [2]. A avaliação do desempenho da iluminação natural tem evoluído para métricas dinâmicas, como a *Daylight Autonomy* (DA) e a *Useful Daylight Illuminance* (UDI), que incorporam a variabilidade diária e sazonal das condições exteriores e permitem uma caracterização mais realista da disponibilidade de luz natural ao longo do tempo [3]. Apesar das suas vantagens, a iluminação natural apresenta limitações intrínsecas, nomeadamente a dependência das condições climáticas e o risco de transferência térmica excessiva por radiação solar direta, que pode contribuir para o sobreaquecimento do espaço interior [1]. Porém é importante realçar que a escolha do produto adequado permite mitigar estas limitações. Relativamente ao risco de transferência térmica excessiva, a fração de radiação que é admitida para o espaço interior varia

significativamente entre produtos, sendo quantificada pelo coeficiente de ganho solar SHGC (*Solar Heat Gain Coefficient*). Tecnologias como revestimentos refletivos foram desenvolvidas especificamente para reduzir este valor e, com ele, o risco de sobreaquecimento [4]. Quanto à dependência das condições climáticas, materiais estáticos como os anteriores mantêm as suas propriedades fixas independentemente da variação das condições exteriores, pelo que a sua eficácia é necessariamente um compromisso entre o desempenho em diferentes estações e orientações. Tecnologias de envidraçado dinâmico, como os vidros eletrocromáticos, permitem ajustar as propriedades óticas e térmicas do material em resposta às condições ambientais, reduzindo este compromisso [4]. A seleção do produto adequado constitui, assim, uma estratégia de mitigação reconhecida na literatura para ambas as limitações.

No domínio da ventilação natural, dois mecanismos físicos condicionam diretamente o caudal de ar renovado e a eficácia do arrefecimento: a ação do vento sobre a envolvente do edifício e o efeito de chaminé, resultante das diferenças de temperatura entre o interior e o exterior. Estratégias que aproveitam estes mecanismos contribuem não só para a redução da temperatura do ar interior, mas também para a melhoria da sensação térmica dos ocupantes através do aumento da velocidade do ar [5]. Em edifícios industriais, a ventilação natural pode ser implementada através de aberturas laterais ou de aberturas de cobertura, sendo o mecanismo físico dominante e o desempenho associado distintos em cada caso. As aberturas laterais, como janelas, portas ou grelhas instaladas em fachadas, podem operar segundo dois modos: a ventilação cruzada (*cross ventilation*), em que o ar entra pela fachada exposta ao vento e sai pela fachada oposta, impulsionado pela diferença de pressão entre ambas, e a ventilação unilateral (*single-sided ventilation*), em que a entrada e a saída de ar ocorrem pela mesma fachada [6]. Esta distinção tem impacto direto no desempenho, dado que medições em ambiente real mostram que a ventilação cruzada garante condições de conforto térmico adequadas durante 70% do tempo de operação, contrastando com apenas 1% no caso da ventilação unilateral [7]. As aberturas de cobertura, por sua vez, exploram preferencialmente o efeito de chaminé, beneficiando do maior diferencial de altura entre a admissão de ar ao nível do piso e a exaustão junto à cobertura, podendo assumir diferentes formas construtivas, como claraboias operáveis, lanternins (*roof monitors*) ou aberturas contínuas ao longo da cumeeira. Em edifícios industriais com cargas térmicas internas elevadas, esta última tipologia tem-se mostrado particularmente eficaz na mitigação do sobreaquecimento: um estudo de modelação dinâmica num armazém industrial com cobertura envidraçada mostrou que a introdução de ventilação natural por abertura na cumeeira reduziu o sobreaquecimento de 17% para cerca de 3% das horas ocupadas acima de 28 °C, apresentando ainda vantagens de segurança relativamente à ventilação por abertura lateral informal, como portas de cais de carga [1].

2.1.1 Quadro normativo de conforto térmico e qualidade do ar

O conforto térmico e a qualidade do ar interior em edifícios são regulados por um conjunto de normas internacionais, entre as quais a EN 16798-1, a EN 15251 e a ISO 7730. Ao comparar estas normas é possível identificar os principais limiares de referência para a conceção de sistemas de controlo ambiental [8]. A concentração de CO₂ no interior é usada como indicador indireto da

qualidade do ar por refletir o equilíbrio entre a taxa de renovação de ar e a ocupação do espaço, quando um determinado limiar é ultrapassado, o conforto e a capacidade cognitiva dos ocupantes são progressivamente afetados. A norma EN 16798-1 organiza este risco em quatro categorias, definidas pela concentração absoluta de CO₂, assumindo uma concentração exterior de referência de aproximadamente 400 ppm, conforme apresentado na Tabela 2.1. A Categoria II, recomendada para edifícios mais recentes, corresponde a um limite de 1200 ppm, valor a partir do qual a degradação do conforto começa a ser perceptível para a maioria dos ocupantes [8]. A mesma norma estabelece faixas de temperatura operativa de 20 a 24 °C no inverno e de 23 a 26 °C no verão para esta categoria, que delimitam a zona de conforto térmico para a generalidade dos ocupantes em edifícios de escritórios e terciários.

Tabela 2.1: Categorias de qualidade do ar interior e conforto térmico segundo a norma EN 16798-1.

Categoria	Descrição	CO₂ (ppm)	Temp. operativa
I	Elevada qualidade	<950	20–24 °C (inv.) / 23–26 °C (ver.)
II	Boa qualidade (recomendada)	<1200	20–24 °C (inv.) / 23–26 °C (ver.)
III	Qualidade aceitável	<1750	18–25 °C (inv.) / 23–27 °C (ver.)
IV	Qualidade inferior	>1750	Abaixo da Categoria III

2.1.2 Estratégias intrínsecas de controlo

Sobre a base destes conceitos físicos, a literatura identifica algumas estratégias de controlo que dependem apenas do próprio sistema e de sensores locais, sem necessidade de coordenação com outros subsistemas do edifício. Uma das mais estudadas é a ventilação noturna (*night cooling*), que consiste na abertura controlada do sistema durante o período noturno, quando a temperatura exterior é inferior à interior, permitindo arrefecer a massa térmica do edifício e criar uma reserva de frio disponível para absorver as cargas térmicas do período de ocupação seguinte [9]. A eficácia desta estratégia depende, no entanto, da amplitude do diferencial térmico disponível, dado que abaixo de um determinado limiar a energia removida pela ventilação deixa de compensar as perdas pela envolvente, tornando a estratégia ineficaz. Artmann et al. [10] analisaram este potencial em 259 estações meteorológicas europeias e estabeleceram um diferencial mínimo de 3 °C entre a temperatura interior e exterior como condição para a eficácia da estratégia, identificando potencial elevado em toda a Europa do Norte e potencial significativo em regiões do centro, leste e sul do continente, incluindo Portugal. Deste modo, a disponibilidade deste diferencial térmico constitui um dos fatores determinantes para o desempenho da ventilação noturna.

Importa, contudo, salientar que esta condição poderá tornar-se progressivamente menos frequente em cenários climáticos futuros. Estudos baseados em simulações de desempenho energético em condições climáticas projetadas indicam uma redução do potencial da ventilação noturna mecânica, devido ao aumento das temperaturas mínimas e à diminuição da amplitude térmica diária em vários cenários de aquecimento global, o que pode comprometer a sua eficácia relativa

enquanto estratégia passiva de arrefecimento [11]. Paralelamente, observações recentes evidenciam já um aumento estatisticamente significativo da frequência e intensidade das noites quentes em regiões de clima subtropical e mediterrânico, fenómeno consistente com as tendências associadas às alterações climáticas e suscetível de reduzir o potencial desta estratégia em regiões de clima semelhante [12].

A nível do controlo das aberturas, a modulação contínua da percentagem de abertura é apontada como uma alternativa mais flexível ao controlo binário (totalmente aberto ou totalmente fechado), por permitir adaptar o caudal de ar renovado às condições ambientais reais em vez de o limitar a dois estados discretos. Esta abordagem justifica o recurso a níveis intermédios de abertura em sistemas de ventilação natural, sendo a base sobre a qual estratégias mais avançadas, como a integração de informação preditiva, podem ser construídas, conforme se discute na Secção 2.2. O impacto energético da automatização da ventilação natural pode ainda ser quantificado com base em estudos de simulação, como é o caso do estudo realizado por Schulze e Eicker [13], onde é demonstrado que a ventilação natural controlada pode reduzir o consumo de arrefecimento entre 13 e 44 kWh/m²-ano, dependendo do clima e da configuração das aberturas. Estes valores confirmam que o ganho energético associado a estratégias passivas devidamente controladas é mensurável e relevante.

2.1.3 Necessidade de coordenação automática

O desempenho efetivo das estratégias descritas depende fortemente da forma como são geridas ao longo do tempo. Em sistemas que integram iluminação e ventilação num mesmo elemento da envolvente, como é o caso do *AirLight*, a ausência de coordenação funcional pode conduzir a conflitos entre objetivos de desempenho distintos, resultando em comportamentos inconsistentes e dependentes do utilizador [7]. A integração física de soluções passivas não é, por si só, suficiente, sendo que a aplicação de estratégias de controlo automático constitui um requisito determinante para a exploração eficaz do seu potencial. Para sistemas com requisitos de coordenação mais alargados, que envolvem conceitos como articulação com climatização, a integração com iluminação artificial e antecipação baseada em previsão meteorológica, esta exigência traduz-se na necessidade de uma camada supervisora dedicada, que se analisa na secção seguinte.

2.2 Sistemas de Gestão de Edifícios

Os Sistemas de Gestão de Edifícios, ou *Building Management Systems* (BMS), são plataformas responsáveis pela supervisão, monitorização e coordenação dos sistemas técnicos de um edifício, incluindo climatização, ventilação, iluminação e segurança. Através da recolha contínua de dados e da gestão centralizada de subsistemas, os BMS permitem implementar estratégias que contribuem para a melhoria da eficiência energética do edifício e do conforto dos ocupantes sem necessidade de intervenção manual constante.

2.2.1 Estratégias supervisoras

Os BMS implementam um conjunto de estratégias supervisoras que coordenam o funcionamento dos diferentes subsistemas técnicos do edifício, tirando partido da sua posição centralizada para integrar informação de múltiplas fontes e tomar decisões que dificilmente seriam possíveis a nível local. Estas estratégias distinguem-se das abordagens descritas na Secção 2.1 pela necessidade de coordenar subsistemas distintos, integrar serviços externos ou aplicar lógicas de antecipação que requerem uma camada supervisora dedicada.

Uma das estratégias mais consolidadas é a ventilação controlada pela procura (*demand-controlled ventilation*), que ajusta a taxa de renovação de ar à ocupação real do espaço, medida indiretamente pela concentração de CO₂, em vez de operar com caudais constantes ou horários fixos. Esta abordagem permite garantir condições adequadas de qualidade do ar interior apenas quando necessário, com base nos limiares normativos apresentados na Secção 2.1, evitando o consumo associado a renovações de ar desnecessárias em períodos de baixa ocupação [14].

A combinação de ventilação natural com sistemas mecânicos de climatização num mesmo edifício é tratada na literatura sob o conceito de ventilação em modo misto (*mixed-mode ventilation*). Esta abordagem reconhece que a ventilação natural não substitui integralmente os sistemas mecânicos em todas as condições climáticas, e que a articulação entre os dois oferece um equilíbrio entre eficiência energética e garantia de conforto. Foi possível estabelecer a classificação destas estratégias em três categorias: *concurrent*, quando ambos os sistemas podem operar simultaneamente na mesma zona com a ventilação natural a complementar a climatização ativa; *changeover*, em que apenas um dos sistemas opera de cada vez na mesma zona, alternando consoante as condições; e *zoned*, em que ambos coexistem mas atuam em zonas distintas do edifício [15].

A relevância prática desta coordenação não decorre apenas dos ganhos energéticos potenciais, mas também do risco associado à sua ausência. A operação simultânea descoordenada, com janelas abertas em paralelo com climatização ativa, resulta em desperdício energético direto, dado que o sistema mecânico procura manter um *setpoint* contra a perda de ar climatizado para o exterior. O impacto energético da implementação de ventilação em modo misto em edifícios residenciais, reporta poupanças de aproximadamente 24% no consumo de climatização durante os meses de transição da primavera e 27% durante os meses de transição do outono [16].

No domínio luminoso, o equivalente à ventilação em modo misto é tratado na literatura sob o conceito de *daylight harvesting*, que consiste em modular o funcionamento da iluminação artificial em função da iluminação natural disponível, evitando manter as luminárias em pleno quando a luz natural é suficiente para atingir o nível de iluminância pretendido. É então possível sistematizar as duas abordagens predominantes neste domínio: o *daylight switching*, em que as luminárias são ligadas ou desligadas consoante a disponibilidade de luz natural ultrapasse ou não um determinado limiar, e o *daylight dimming*, em que o fluxo luminoso das luminárias é regulado de forma contínua para complementar a luz natural até atingir o nível pretendido [17]. A integração com sensores de ocupação é, em ambos os casos, frequentemente considerada como camada adicional que evita o consumo em espaços desocupados.

A mesma revisão identifica um conjunto de desafios práticos que condicionam a eficácia destas estratégias, nomeadamente a robustez técnica dos sistemas, a integração arquitetural com o edifício e a aceitação por parte dos utilizadores. Quantitativamente, é reportado um potencial de poupança entre 20% e 60% no consumo de iluminação face a sistemas sem regulação, dependendo da estratégia adotada e do contexto de aplicação [17]. Estudos focados no controlo integrado de iluminação e HVAC em edifícios públicos confirmam estes valores, revelando reduções de consumo energético de iluminação até 39% face ao controlo manual, num caso em que a estratégia combinava *daylight harvesting* com controlo por ocupação [18].

Uma estratégia de supervisão que tem ganho relevância em sistemas BMS modernos é o controlo antecipativo baseado em previsão meteorológica. Em vez de reagir apenas aos sensores no momento, o sistema consulta serviços externos de previsão, tipicamente APIs disponibilizadas por serviços meteorológicos, e ajusta o funcionamento dos diferentes subsistemas em função das condições previstas para as horas seguintes. Esta capacidade permite, por exemplo, fechar aberturas antes de uma precipitação prevista, antecipar a ventilação em períodos de calor extremo, ou ajustar o funcionamento da climatização para tirar partido de períodos de temperatura exterior favorável.

Em [19], é possível encontrar uma revisão do estado da arte sobre controlo preditivo em sistemas HVAC, identificam a integração de dados de previsão meteorológica como uma das direções consolidadas na investigação em automação predial, sendo já uma funcionalidade presente em plataformas BMS comerciais. No domínio específico da ventilação natural, é demonstrado concretamente o valor desta abordagem ao propor uma estratégia em que a percentagem de abertura é modulada continuamente entre 0% e 100% em função da temperatura operativa interior, combinada com previsão da temperatura exterior para antecipar condições adversas [20]. Os resultados mostram que a integração da previsão melhora a manutenção do conforto térmico interior, reduz o efeito de histerese e evita situações de sobre-ventilação, comparativamente a estratégias puramente reativas.

2.2.2 Quadro normativo automação predial

O impacto da automação na eficiência energética dos edifícios é enquadrado pela norma EN 15232, que estabelece uma classificação em quatro classes, de A (elevada automação) a D (sem automação eficiente), tendo a classe C como referência. Para cada classe, a norma define fatores de eficiência energética que funcionam como multiplicadores do consumo de energia térmica e elétrica face à classe de referência. Um fator inferior a 1 indica uma redução de consumo, enquanto um fator superior indica um consumo acrescido. A Tabela 2.2 apresenta os fatores definidos pela norma para edifícios de escritórios, ilustrando que um sistema de classe A pode reduzir o consumo de energia térmica em até 30% face à classe de referência [14].

Tabela 2.2: Fatores de eficiência energética segundo a norma EN 15232.

Classe	Descrição	Fator térmico	Fator elétrico
A	Elevada automação e gestão técnica	0.70	0.85
B	Automação avançada	0.88	0.93
C	Automação standard (referência)	1.00	1.00
D	Sem automação eficiente	1.51	1.10

2.2.3 Limitações e relação com controladores locais

Os BMS operam tipicamente como sistemas de supervisão de alto nível, sem garantias temporais estritas nem comportamento determinístico. Esta característica limita a sua adequação para funções de controlo em tempo real ou de segurança, que devem ser tratadas com maior proximidade ao equipamento. Os BMS são, por isso, utilizados sobretudo para monitorização e coordenação global, complementados por controladores locais, como os autómatos programáveis, responsáveis pela execução das lógicas de controlo determinístico. Adicionalmente, importa que os parâmetros definidos no BMS estejam alinhados com os limiares definidos pelas normas internacionais de qualidade ambiental, garantindo que as decisões automáticas tomadas pelo sistema correspondem a critérios reconhecidos de conforto e bem-estar dos ocupantes [8]. Esta separação funcional sustenta a adoção de arquiteturas hierárquicas, nas quais o BMS atua como camada supervisora de alto nível e os controladores locais asseguram a execução das lógicas em tempo real, modelo este que está na base da arquitetura desenvolvida no presente projeto [14].

2.3 Autómatos Programáveis e Protocolos de Comunicação

Os autómatos programáveis industriais, habitualmente designados por *Programmable Logic Controllers* (PLCs), são amplamente referidos na literatura como elementos centrais para a implementação de controlo local em sistemas de automação de edifícios, particularmente em aplicações que exigem integração direta com sensores e atuadores. A sua principal vantagem reside na capacidade de operarem diretamente sobre sinais de entrada e saída industriais, com tempos de resposta previsíveis e independentes das camadas superiores de supervisão, tornando-os adequados para a execução de lógicas de controlo determinístico em tempo real. No contexto da automação de edifícios, os PLCs são frequentemente utilizados para o controlo de sistemas HVAC, ventilação e iluminação, executando algoritmos baseados em variáveis ambientais e estados operacionais [21]. Os PLCs são tipicamente programados em linguagens normalizadas pela norma IEC 61131-3, que define cinco linguagens de programação: *Ladder Diagram* (LD), *Function Block Diagram* (FBD), *Structured Text* (ST), *Instruction List* (IL) e *Sequential Function Chart* (SFC). Esta normalização facilita a implementação de estratégias de controlo modulares, a reutilização de código e a manutenção do sistema ao longo do seu ciclo de vida [21]. A possibilidade de integração dos PLCs com sistemas de supervisão, como plataformas SCADA ou BMS, permite ainda a monitorização do estado do sistema e a intervenção por parte do operador, complementando o controlo local com uma camada de gestão global.

Desta forma, a escolha do protocolo de comunicação influencia diretamente a latência, a fiabilidade, a interoperabilidade entre dispositivos de diferentes fabricantes e a facilidade de integração com sistemas externos, sendo por isso uma decisão estruturante na concepção da arquitetura do sistema [14, 22]. No contexto do presente trabalho, destacam-se dois protocolos relevantes: o Modbus, amplamente utilizado em automação industrial e de edifícios, e o REST sobre HTTP, característico de arquiteturas modernas de IoT e de aplicações web.

2.3.1 Modbus

O *Modbus* é um protocolo de comunicação aberto e amplamente adotado em sistemas de automação industrial e de edifícios. Segue um modelo *master-slave*, no qual um dispositivo central, o *master* inicia todas as transações de comunicação e os dispositivos remotos, os *slaves*, respondem a pedidos de leitura ou escrita sobre um conjunto de registos bem definidos [23]. As suas duas variantes principais são: Modbus RTU, que atua sobre interfaces série como o RS-485, e o Modbus TCP, que atua sobre redes Ethernet.

Estas duas variantes partilham o mesmo modelo de dados, diferindo apenas no meio de transporte. A Figura 2.1 ilustra uma arquitetura típica de rede Modbus, em que dispositivos ligados por Modbus TCP sobre Ethernet coexistem com dispositivos em barramentos série (RS-485, RS-232) através de *gateways* de protocolo.

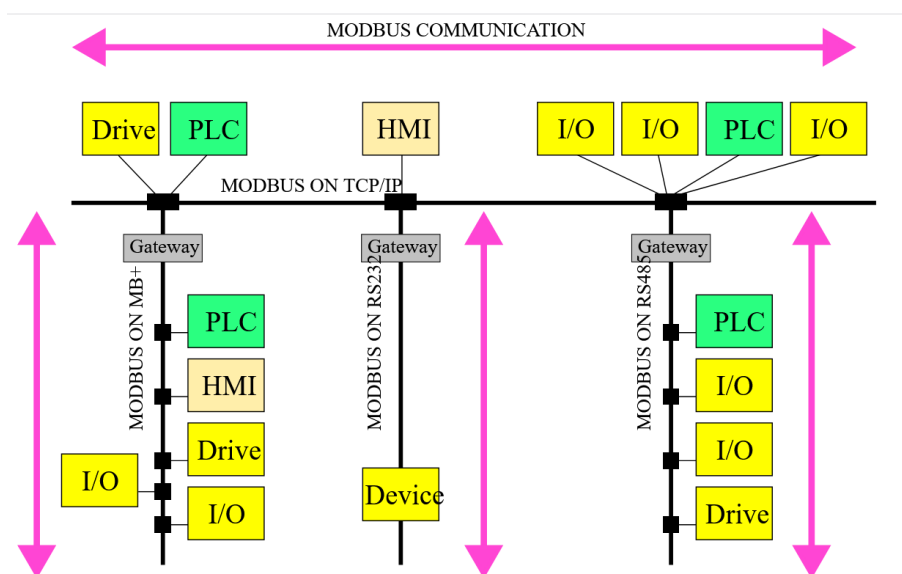


Figura 2.1: Exemplo de arquitetura de rede *Modbus*. Adaptado de [24].

A informação trocada num sistema Modbus é organizada em quatro tipos de registos, distinguidos pelo tipo de dados e pelo sentido de acesso. Por um lado, os *Discrete Inputs* representam valores booleanos só de leitura, tipicamente associados a entradas digitais, enquanto que os *Input Registers* são valores também de leitura, porém de 16 bits e frequentemente utilizados para representar leituras de sensores analógicos. Os *Coils* são valores booleanos de leitura e escrita,

tipicamente associados a saídas digitais, enquanto que os *Holding Registers* são valores de 16 bits de leitura e escrita, usados para parâmetros de configuração e variáveis de processo. Para cada dispositivo *slave* é-lhe atribuído um endereço único (*Unit ID*), permitindo a coexistência de múltiplos dispositivos na mesma rede.

A comunicação entre *master* e *slave* é estruturada em funções normalizadas, cada uma identificada por um código (*function code*) e associada a um tipo específico de operação sobre os registos do dispositivo. As funções de leitura permitem ao *master* obter o estado dos registos do *slave*, enquanto as funções de escrita permitem alterar o valor de um ou mais registos. A tabela 2.3 resume as funções mais frequentemente utilizadas, agrupando-as por tipo de operação e indicando o tipo de registo a que se aplicam.

Tabela 2.3: Funções Modbus.

Código	Função	Operação
01	<i>Read Coils</i>	Leitura múltipla de <i>Coils</i>
02	<i>Read Discrete Inputs</i>	Leitura múltipla de <i>Discrete Inputs</i>
03	<i>Read Holding Registers</i>	Leitura múltipla de <i>Holding Registers</i>
04	<i>Read Input Registers</i>	Leitura múltipla de <i>Input Registers</i>
05	<i>Write Single Coil</i>	Escrita única em <i>Coil</i>
06	<i>Write Single Register</i>	Escrita única em <i>Holding Register</i>
15	<i>Write Multiple Coils</i>	Escrita múltipla em <i>Coils</i>
16	<i>Write Multiple Registers</i>	Escrita múltipla em <i>Holding Registers</i>

A simetria da tabela reflete a estrutura interna do protocolo, onde cada tipo de registo dispõe de funções dedicadas para leitura e, no caso dos registos editáveis (*Coils* e *Holding Registers*), funções para escrita única ou múltipla. Os registos só de leitura (*Discrete Inputs* e *Input Registers*) não dispõem de funções de escrita, refletindo a sua natureza de variáveis observáveis, mas não modificáveis pelo *master*.

Na prática industrial, o *Modbus* é frequentemente utilizado para a integração de dispositivos heterogéneos numa mesma rede. Numa topologia típica, um único *master*, geralmente um BMS, um sistema SCADA ou um PLC de nível superior, é responsável por coordenar a comunicação com múltiplos *slaves* que incluem outros PLCs, sensores, atuadores, painéis HMI e *gateways* para outros protocolos [22]. Cada *slave* expõe um mapa de registos próprio, documentado pelo fabricante, que cabe ao integrador mapear para variáveis lógicas do sistema de supervisão. As duas variantes do protocolo coexistem habitualmente no mesmo edifício ou instalação, ocupando níveis distintos da hierarquia de comunicação [14, 23].

O Modbus RTU é utilizado ao nível de campo, sobre interfaces série como o RS-485, em topologias do tipo *daisy-chain* onde múltiplos dispositivos partilham o mesmo barramento físico e são identificados por endereços únicos entre 1 e 247. A comunicação ocorre em modo *half-duplex*, ou seja, apenas um dispositivo pode transmitir de cada vez.

O Modbus TCP, por sua vez, é utilizado ao nível superior da hierarquia, sobre redes Ethernet, sendo a forma mais comum de integrar PLCs e equipamentos de automação em plataformas de supervisão centralizada como BMS ou SCADA [22]. Cada *slave* passa a ser identificado pelo seu

endereço IP em vez de um endereço de barramento, e a comunicação utiliza a porta TCP/IP padronizada 502. O protocolo beneficia das características da pilha TCP/IP, nomeadamente a detecção de erros e a entrega ordenada de pacotes. A topologia em estrela típica das redes Ethernet permite a coexistência de múltiplos *masters* acedendo simultaneamente aos mesmos *slaves*, característica que não é viável na variante RTU. Em contrapartida, o Modbus TCP apresenta uma maior complexidade de configuração de rede e exige a definição de políticas de segurança quando exposto a redes externas, dado que o protocolo não inclui mecanismos nativos de autenticação ou cifra [23].

O Modbus caracteriza-se por uma latência previsível e por um comportamento determinístico, particularmente na sua variante RTU, o que o torna adequado para aplicações industriais com requisitos de previsibilidade temporal [22]. A sua simplicidade de implementação e o facto de ser um protocolo aberto e sem licenciamento contribuem para a sua ampla adoção em sistemas HVAC, BMS industriais, PLCs e equipamentos de campo. Entre as suas limitações encontram-se a ausência de um modelo de dados padronizado e a inexistência de funcionalidades nativas de segurança, como autenticação ou cifra, o que requer cuidados adicionais quando o protocolo é exposto a redes externas.

2.3.2 REST sobre HTTP

A arquitetura REST (*Representational State Transfer*) sobre HTTP segue um modelo cliente-servidor em que os recursos do sistema são identificados por URLs e manipulados através de métodos HTTP padronizados [25]. O conceito central da arquitetura é o de recurso, onde qualquer entidade do sistema, seja uma variável, um dispositivo ou uma coleção de dados, é representada como um recurso identificado de forma unívoca por um URL, sobre o qual o cliente pode realizar operações. Esta representação uniforme dos dados expostos pelo sistema é um dos fatores que confere ao REST a sua simplicidade conceptual e a sua facilidade de adoção.

A comunicação é *stateless*, o que significa que cada pedido contém toda a informação necessária para ser interpretado e respondido, sem dependência de estado armazenado entre pedidos sucessivos. Esta propriedade permite simplificar a implementação tanto do servidor como dos clientes e permite que diferentes clientes interajam com o mesmo servidor sem interferência mútua [25]. Diferentemente do Modbus, o REST não foi concebido especificamente para automação industrial, mas tornou-se progressivamente relevante neste domínio devido à crescente integração de sistemas de edifícios com aplicações web, móveis e serviços *cloud* característicos do paradigma IoT.

A interação entre cliente e servidor numa API REST é estruturada em torno de um conjunto de métodos HTTP padronizados, cada um com uma semântica específica. O método GET é utilizado para obter o estado atual de um recurso identificado por um URL, sem alterar o estado do servidor, sendo, portanto, idempotente, no sentido em que múltiplas invocações sucessivas produzem sempre o mesmo resultado. O método POST é utilizado para enviar dados ao servidor com o objetivo de criar, atualizar ou desencadear uma ação sobre um recurso. Este método, por sua vez, não é idempotente, dado que cada invocação pode produzir efeitos distintos. O padrão inclui ainda os métodos PUT, para a atualização completa de um recurso existente, e DELETE, para a

remoção de um recurso, embora a sua utilização dependa do desenho de cada API e nem todas as implementações os exponham. A escolha do método apropriado em cada operação é uma decisão importante na conceção de uma API REST, dado que afeta tanto o comportamento esperado pelos clientes como a forma como a operação é tratada por elementos intermédios da rede. O resultado de cada pedido é comunicado ao cliente através de um código de estado HTTP, organizado em categorias que indicam a natureza da resposta. Os códigos da família 2xx indicam sucesso, sendo o 200 (OK) o mais frequente. Já os da família 4xx indicam erros do lado do cliente, como pedidos malformados (400) ou recursos inexistentes (404), por fim, os da família 5xx indicam erros internos do servidor (500). Esta categorização permite ao cliente reagir de forma diferenciada ao resultado de cada pedido sem necessidade de interpretar o conteúdo detalhado da resposta.

O corpo das mensagens trocadas é tipicamente formatado em JSON (*JavaScript Object Notation*), um formato textual leve baseado em pares chave-valor e nativamente suportado pela generalidade das linguagens de programação modernas, o que contribuiu decisivamente para a sua adoção como formato dominante na troca de dados em arquiteturas deste tipo. A principal vantagem do REST/HTTP é a sua universalidade, dado que qualquer dispositivo com pilha TCP/IP pode atuar como cliente ou servidor sem necessidade de *middleware* dedicado ou bibliotecas proprietárias [26]. Esta característica facilita a integração com interfaces de utilizador e aplicações de alto nível desenvolvidas em qualquer linguagem ou *framework*, sendo por isso particularmente adequada para sistemas que exponham dados e comandos a clientes externos como navegadores web, aplicações móveis ou plataformas de supervisão *cloud*. Diferentemente do Modbus, o REST/HTTP herda da pilha TCP/IP um conjunto de mecanismos de segurança maduros, nomeadamente o TLS (*Transport Layer Security*), que dá origem ao HTTPS, e diversos esquemas de autenticação baseados em *tokens* ou credenciais, podendo ser ativados consoante os requisitos de cada implementação e reduzindo significativamente os cuidados adicionais necessários quando a comunicação se faz sobre redes potencialmente expostas [26].

2.3.3 Critérios de escolha e contextos de aplicação

A escolha entre Modbus e REST/HTTP não deve ser entendida como uma comparação direta entre alternativas equivalentes, mas como a seleção do protocolo adequado ao domínio em que vai operar [26]. Cada um dos protocolos é o resultado de uma evolução em contextos distintos e possui o seu próprio domínio de excelência. O Modbus é o protocolo natural para integração entre equipamentos industriais que partilham o mesmo ecossistema, sejam PLCs, sistemas HVAC, BMS industriais e equipamentos de campo, onde a latência previsível, o determinismo e a robustez face a condições adversas são requisitos críticos. O REST/HTTP, por sua vez, é o protocolo natural para exposição de dados e comandos a aplicações de alto nível, como interfaces web, aplicações móveis e serviços *cloud*, onde a universalidade da pilha TCP/IP e a independência de *middleware* são vantagens decisivas. Esta divisão de papéis significa que, num sistema completo de automação de edifícios, ambos os protocolos podem coexistir e ser usados em paralelo, cada um no seu domínio de aplicação.

Atualmente têm surgido abordagens híbridas, em que os dois tipos de protocolos coexistem num mesmo sistema [25, 27]. Nestas abordagens, o Modbus é utilizado para a comunicação local entre o controlador e os dispositivos industriais ou sistemas BMS, enquanto o REST/HTTP é utilizado para expor dados e comandos a aplicações de alto nível, como interfaces web, aplicações móveis ou plataformas de supervisão cloud. Esta combinação permite tirar partido das vantagens de cada protocolo no domínio em que se mostram mais adequados, sendo particularmente característica de sistemas IoT aplicados a edifícios inteligentes [25].

2.4 O Sistema *AirLight*

O *AirLight*, representado na Figura 2.2, surge então como um equipamento que integra iluminação natural difusa e ventilação natural num único sistema físico aplicado em cobertura. Procura tirar partido das estratégias passivas, cuja eficácia depende, como visto, da existência de um sistema de controlo adequado. Neste equipamento a iluminação difusa é captada através de uma cúpula piramidal, que maximiza a área de captação. A geometria e a camada de policarbonato alveolar da cúpula são responsáveis pela componente ótica do sistema, eliminando a incidência solar direta e filtrando a componente infravermelha da radiação incidente, contribuindo para um isolamento térmico superior ao das claraboias convencionais. Um vidro temperado laminado, integrado na constituição do equipamento, assegura ainda a resistência do equipamento à propagação de fogo exterior através da cobertura, característica particularmente relevante em edifícios industriais, onde claraboias convencionais em policarbonato sem esta classificação constituem pontos vulneráveis de propagação de incêndio em caso de fogo exterior.

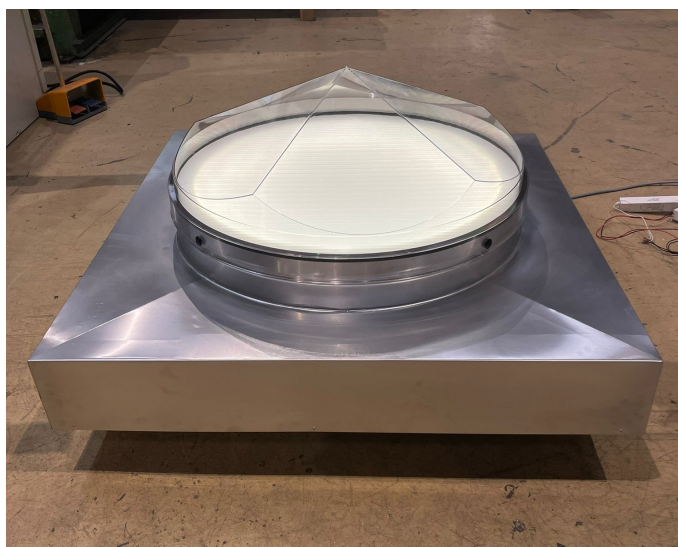


Figura 2.2: Sistema *AirLight*.

O movimento de abertura é elevatório e acionado por um atuador elétrico, cujo funcionamento é detalhado no Capítulo 3. A estrutura superior do equipamento, onde se encontra a cúpula, eleva-se em relação à base fixa instalada na cobertura, guiada por um conjunto de barras metálicas

dispostas em tesoura que se estendem progressivamente com a subida, até um curso máximo de abertura de aproximadamente 40 cm. Esta elevação expõe um vão lateral perimetral, por onde o ar é admitido e expelido. A estrutura superior, ao manter-se sobre a abertura durante todo o curso de movimento, protege o vão contra a incidência direta de chuva e a entrada de elementos indesejados, ao contrário das claraboias convencionais cuja abertura expõe diretamente o espaço interior ao exterior. Na posição fechada, a estrutura desce sobre a base, cobrindo as guias e eliminando a abertura lateral. Na Figura 2.3 é possível visualizar o sistema numa posição de abertura.



Figura 2.3: *AirLight* na posição aberta.

Do ponto de vista das estratégias revistas ao longo deste capítulo, o *AirLight* tira partido natural de várias delas pela sua própria conceção física. A iluminação natural difusa que proporciona insere-se diretamente no domínio em que o *daylight harvesting* encontra aplicação, dado que substitui parcialmente a iluminação artificial em períodos de luz natural suficiente. O seu difusor, ao eliminar a incidência solar direta e filtrar a componente infravermelha, reduz ainda o risco de sobreaquecimento discutido na Secção 2.1, de forma análoga ao papel desempenhado pelos revestimentos de controlo solar referidos [4]. Também a ventilação natural por abertura em cobertura aproveita o efeito de chaminé, sendo a abertura elevatória, uma solução particular desta tipologia. A capacidade de assumir posições intermédias de abertura abre, por sua vez, caminho à aplicação de estratégias de modulação contínua, ao invés do controlo binário.

Por outro lado, existem estratégias que dependem inteiramente da existência de uma camada de controlo adequada e não decorrem apenas da conceção física do equipamento, exigindo que o sistema de controlo as implemente explicitamente, seja a nível local, ou através de uma camada supervisora. É o caso da ventilação noturna, estratégia que o *AirLight* pode também explorar para fins de *free-cooling*, dada a sua capacidade de abertura controlada. A coordenação com sistemas mecânicos de climatização e a antecipação baseada em previsão meteorológica, são também

estratégias que o sistema pode explorar. Do ponto de vista funcional, o comportamento do equipamento e as estratégias de controlo possíveis são ainda condicionados pela cinemática de abertura e pelas características do atuador elétrico responsável pelo movimento elevatório. A existência de posições intermédias permite uma exploração mais flexível das funcionalidades de ventilação e iluminação, mas exige uma lógica de controlo adaptada e validada para gerir a relação entre a posição de abertura e o efeito pretendido.

A integração de iluminação natural e ventilação num mesmo elemento da envolvente não é, contudo, uma proposta isolada no mercado. Existem soluções comerciais com objetivos semelhantes, frequentemente desenvolvidas em paralelo com sistemas de desenfumagem por partilharem o mesmo princípio físico de abertura em cobertura. A análise destas soluções, bem como o enquadramento normativo dos sistemas de desenfumagem com função combinada de conforto, permite posicionar o *AirLight* face ao estado da arte atual.

2.5 Sistemas de Desenfumagem

A necessidade de integrar funções de ventilação natural e desenfumagem num único sistema físico aplicado em cobertura tem conduzido ao desenvolvimento de soluções designadas por *dual purpose* ou, na terminologia técnica europeia, *dual function ventilation*. Nestes sistemas, o mesmo equipamento serve duas funções com prioridades distintas: ventilação natural de conforto e extração de fumo em situação de incêndio. O conceito está formalmente descrito na patente europeia EP 2003262 A2 [28], que estabelece o princípio de uma abertura única usada para os dois cenários.

A série de normas EN 12101 regula os sistemas de controlo de fumo e calor em edifícios europeus. Duas partes desta série são particularmente relevantes neste contexto: a EN 12101-2, que define os requisitos para os ventiladores naturais de desenfumagem aplicáveis a equipamentos do tipo estudado, e a EN 12101-9, que define os requisitos para os painéis de controlo, sendo a norma a que obedece a central considerada no estudo de uma possível coexistência, apresentado nos Capítulos, 3 e 4. Em ambos os casos, é imposta uma separação clara entre modos de operação e garantida a prioridade absoluta da função de desenfumagem em situação de emergência.

Subjacente a este quadro normativo está um princípio comum na literatura técnica e nos documentos normativos europeus: as funções de segurança contra incêndio permanecem em equipamento certificado e na sua cadeia dedicada, podendo sistemas não certificados, como BMS ou sistemas de controlo externos, coexistir com eles e ler o seu estado, mas nunca comandar nem interferir com a função de segurança [22, 27]. Esta separação é uma condição para a integração de sistemas com diferentes níveis de criticidade no mesmo edifício, e está na base das opções arquiteturais adotadas pelos fabricantes de sistemas *dual function* disponíveis no mercado, analisados nas secções seguintes.

2.5.1 Análise de Soluções Comerciais

Entre as soluções comerciais analisadas, o modelo CSP da empresa VELUX destaca-se como um exemplo de elevada maturidade ao nível do sistema de controlo. Trata-se de uma claraboia de

cobertura com abertura motorizada, concebida para permitir a entrada de luz natural e a ventilação diária, bem como a abertura total em situações de emergência para efeitos de desenfumagem. A distinção entre ventilação de conforto e abertura de emergência é suportada por um sistema de controlo dedicado e proprietário, fornecido pelo fabricante, que gere os diferentes modos de operação e assegura a prioridade da função de segurança. No mercado atual, esta solução caracteriza-se como a mais semelhante ao produto *AirLight*, devido ao seu movimento também de abertura vertical.

Em termos de estratégias de controlo, a VELUX disponibiliza, em complemento ao controlador base, o sistema *VELUX Active*, que adiciona uma camada supervisora de gestão climática. Este sistema integra sensores de temperatura, humidade e CO₂, com base nos quais opera automaticamente a abertura, implementando, na prática, uma estratégia de ventilação controlada pela procura. Estão também previstas funcionalidades como o fecho automático em caso de precipitação, através de sensor de chuva dedicado, e a antecipação baseada em previsão meteorológica, que ajusta o funcionamento dos sistemas de proteção solar e ventilação em função das condições previstas para o dia. Estas funcionalidades reproduzem, num produto comercial, várias das estratégias de supervisão discutidas anteriormente.

Do ponto de vista da arquitetura de comunicação, a VELUX adota uma abordagem centralizada em controladores proprietários, sendo a comunicação com sistemas BMS externos realizada de forma indireta através de *gateways* dedicados. Estes *gateways* funcionam como interfaces de tradução entre o protocolo interno do fabricante e protocolos normalizados como BACnet ou KNX, realizando um mapeamento estático entre comandos internos e pontos externos. A lógica de controlo permanece totalmente encapsulada no controlador principal, não sendo acessível nem configurável externamente, o que garante a preservação do comportamento determinístico e das funções de segurança, mas limita a profundidade da integração. O BMS interage, assim, com uma representação simplificada do sistema, limitada a comandos globais e estados agregados, reduzindo a flexibilidade e a granularidade dos pontos expostos.

O equipamento *STEEL MOE* da empresa D+H adota uma abordagem distinta, baseada numa arquitetura mais aberta. Este sistema consiste numa claraboia metálica equipada com atuadores elétricos, permitindo a abertura do vão em cobertura tanto para ventilação natural como para extração de fumos em situação de incêndio.

Quanto às estratégias de controlo, os controladores da família D+H incorporam funções de conforto que vão para além do acionamento manual das aberturas. Os transmissores de vento e chuva (*wind/rain transmitter*) asseguram o fecho automático em condições meteorológicas adversas, e os módulos de controlo de temperatura permitem a ventilação automática dependente da temperatura interior. Modelos mais avançados, como o controlador CPS-M com o módulo analógico-digital, suportam ainda a integração de sensores de CO₂ e temperatura para implementação de ventilação controlada pela procura. A documentação técnica refere explicitamente a possibilidade de configurar estratégias de ventilação noturna (*night cooling*) para arrefecimento da massa térmica do edifício. À semelhança da VELUX, o conjunto de estratégias disponibilizadas corresponde ao que a literatura identifica como práticas estabelecidas em sistemas de automação

de edifícios.

A lógica de controlo é separada do equipamento físico, sendo implementada em painéis de controlo SHEV nos quais os protocolos Modbus RTU/TCP e BACnet/IP são disponibilizados diretamente. Esta abordagem confere ao sistema maior adaptabilidade e facilidade de integração em ambientes BMS, embora exija um maior esforço de conceção e validação da lógica de controlo. No caso do Modbus, o painel atua como *server*, disponibilizando um mapa de registos fixo e documentado, no qual cada registo representa comandos de alto nível, estados do sistema ou alarmes. A escrita em registos é tipicamente condicionada por estados internos do painel, impedindo que comandos externos violem prioridades funcionais ou requisitos de segurança. A implementação de BACnet/IP introduz um nível adicional de estruturação, através da exposição de objetos normalizados como *Binary Input*, *Binary Output* e *Multi-State Objects*, permitindo ao BMS interagir com o sistema de forma semanticamente consistente. Em ambos os casos, o conjunto de pontos disponibilizados é limitado e pré-definido pelo fabricante, assegurando que o controlo externo não interfere com a lógica crítica de segurança e que a função de desenfumagem permanece protegida.

2.5.2 Padrões Arquiteturais

A análise das soluções comerciais permite identificar dois padrões arquiteturais distintos. O primeiro, representado pelo VELUX CSP, caracteriza-se por um controlo fortemente integrado e proprietário, que oferece robustez e simplicidade de instalação à custa de menor flexibilidade de integração com sistemas externos. O segundo, representado pelo D+H STEEL MOE, privilegia uma arquitetura aberta na qual a lógica de controlo é separada do *hardware*, permitindo maior adaptabilidade e integração em ambientes BMS, mas exigindo maior esforço de desenvolvimento.

A análise das soluções comerciais permite identificar dois padrões arquiteturais distintos. O primeiro, representado pelo VELUX CSP, caracteriza-se por um controlo fortemente integrado e proprietário, que oferece robustez e simplicidade de instalação à custa de menor flexibilidade de integração com sistemas externos. O segundo, representado pelo D+H STEEL MOE, privilegia uma arquitetura aberta na qual a lógica de controlo é separada do *hardware*, permitindo maior adaptabilidade e integração em ambientes BMS, mas exigindo maior esforço de desenvolvimento.

2.6 Síntese do Enquadramento

A revisão da literatura apresentada neste capítulo permitiu identificar três dimensões fundamentais para o problema abordado nesta dissertação. A primeira prende-se com a utilização de estratégias passivas de iluminação e ventilação natural, cujo contributo para a eficiência energética e para o conforto dos ocupantes se encontra amplamente documentado. Contudo, a sua exploração eficaz depende da implementação de lógicas de controlo capazes de determinar quando, em que medida e em função de que condições o sistema deve operar. A segunda dimensão corresponde à supervisão, na qual os sistemas de gestão técnica de edifícios coordenam estratégias como a ventilação noturna, a ventilação controlada pela procura, a articulação com sistemas de climatização e a antecipação de condições meteorológicas, recorrendo a uma camada de decisão situada

acima do controlador local. A terceira dimensão refere-se à comunicação, onde a coexistência de protocolos industriais, como o Modbus, e de arquiteturas baseadas em REST/HTTP possibilita simultaneamente a integração com equipamentos de campo e a disponibilização de informação a aplicações de nível superior.

A análise das soluções comerciais VELUX CSP e D+H STEEL MOE evidenciou uma arquitetura comum, assente na separação entre a execução física, a lógica de controlo e a comunicação com sistemas externos, diferenciando-se essencialmente pelo grau de abertura e interoperabilidade oferecido. Esta organização constituiu uma referência para a definição da arquitetura proposta neste trabalho.

À data de início desta dissertação, e conforme referido na 1.2, o *AirLight* não dispunha de uma lógica de controlo estruturada, de interfaces de comunicação que permitissem a sua integração com sistemas externos, nem de mecanismos de supervisão. Neste contexto, o trabalho desenvolvido procurou colmatar estas limitações através da conceção de uma arquitetura de controlo inspirada nos princípios identificados na literatura e nas soluções comerciais analisadas, adaptando-os aos requisitos da empresa e às características específicas do equipamento.

Capítulo 3

Sistema e Componentes

3.1 Visão Geral do Sistema

O sistema de controlo do *AirLight* desenvolvido no âmbito desta dissertação assenta numa arquitetura centrada num controlo local e programável, que assume o papel de responsável pela execução da lógica de acionamento dos atuadores. O controlo do sistema pode ser exercido pelo utilizador localmente, através de botões físicos, ou de forma remota, através de uma interface web e de uma aplicação móvel. Paralelamente, o sistema foi desenhado para responder de forma autónoma com base em regras definidas, vindas da própria lógica ou até mesmo de sistemas de supervisão. Em qualquer dos casos, seja manualmente ou de forma automática, o fluxo de controlo converge no dispositivo de controlo local, que interpreta os comandos recebidos e atua sobre a alimentação dos motores responsáveis pela abertura e fecho do sistema, conforme indica a Figura 3.1.

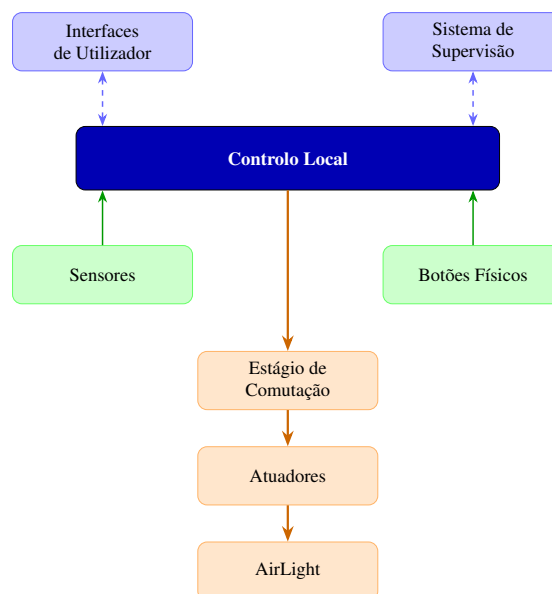


Figura 3.1: Diagrama de blocos geral do sistema.

3.2 Atuadores e Fins de Curso

Na Figura 3.2, está representado o atuador utilizado para o movimento de abertura e fecho do sistema, que se trata de um dispositivo eletromecânico de corrente contínua, do tipo *DC chain window opener*, que converte a energia elétrica num movimento linear de tração. Este movimento é transmitido através de uma corrente metálica articulada, responsável por elevar verticalmente o módulo superior do equipamento segundo uma cinemática elevatória. Os atuadores são alimentados a 24 VDC e o sentido do movimento é determinado pela polaridade aplicada aos terminais de alimentação. A escolha da tensão de alimentação deve-se ao facto de este valor ser tipicamente adotado em atuadores de sistemas de desenfumagem, por assegurar a abertura do equipamento mesmo em caso de falha da rede elétrica. Este aspeto é particularmente relevante para equipamentos que, como o *AirLight*, podem ser integrados em sistemas de desenfumagem certificados nos termos da norma EN 12101-2. A inversão do sentido de deslocamento é obtida através de uma inversão da polaridade da tensão aplicada, que necessita de um estágio intermédio de comutação, responsável por assegurar a correta alimentação do motor em cada modo de operação.

De acordo com a *datasheet* do atuador, as suas principais características técnicas são:

- Força de empurrar/puxar: 400 N;
- Força de bloqueio: 2000 N;
- Curso de deslocamento: 100–1000 mm (configurável);
- Tensão nominal: 24 VDC; corrente nominal 1.2 A; potência 28.8 W;
- Velocidade de operação: 10–15 mm/s;
- Grau de proteção: IP30;
- Funções adicionais: proteção contra sobrecarga e controlo elétrico direto.

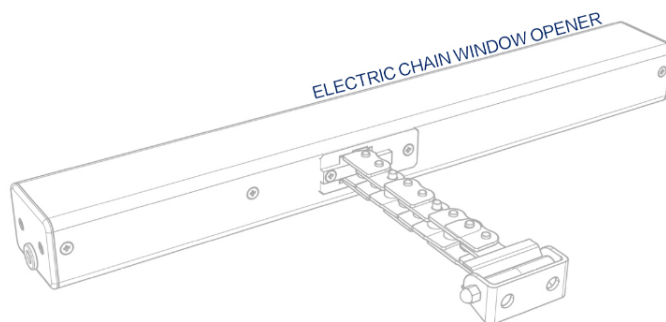


Figura 3.2: Atuador eletromecânico.

Com o objetivo de garantir a segurança mecânica do sistema e evitar danos nos atuadores em situações de excesso de curso, foram instalados fins-de-curso nos limites do movimento de abertura e fecho, como é possível visualizar na Figura 3.3. Os fins-de-curso são interruptores mecânicos acionados pelo próprio movimento do equipamento, ou seja, quando o mecanismo atinge o limite superior ou inferior do seu curso, este é fisicamente premido, interrompendo a alimentação do atuador independentemente do comando aplicado. Esta atuação ocorre diretamente sobre o circuito do atuador, sem envolver o controlador, constituindo uma camada de proteção mecânica autónoma que assegura o funcionamento seguro do sistema mesmo em caso de falha lógica ou de comando.

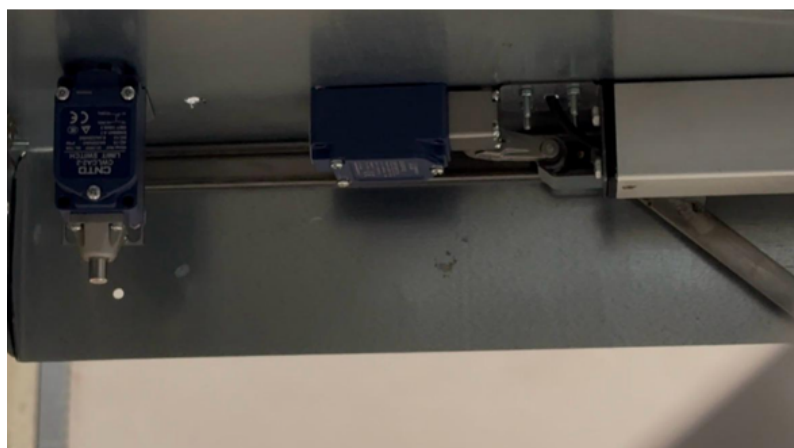


Figura 3.3: Fins de curso instalados.

3.3 Controlo Local

Dado que o *AirLight* se destina sobretudo a aplicações em edifícios industriais, pretendia-se utilizar um controlador com robustez adequada a este tipo de ambiente, que suportasse programação em linguagens normalizadas pela norma IEC 61131-3, permitindo a implementação de uma lógica através de máquinas de estados que estruturassem a gestão da abertura e fecho do sistema. O controlador deveria dispor de entradas digitais e analógicas para integração com sensores ambientais e outros equipamentos, bem como saídas para o acionamento de atuadores. Previa-se a incorporação de conectividade de rede no dispositivo para viabilizar o controlo remoto do sistema via interfaces digitais. Foi igualmente estabelecido como requisito que o controlador selecionado suportasse protocolos de comunicação normalizados, como o *Modbus*, com vista a uma integração robusta com outros sistemas.

Para esta seleção, foram consideradas, por sugestão da empresa, soluções baseadas em micro-controladores de uso genérico, como o ESP32, e em sistemas de automação residencial simplificados, como o Shelly Pro 2PM. O ESP32, apesar da sua flexibilidade e baixo custo, não dispõe de certificação de robustez industrial equivalente à de um PLC dedicado, o que o tornava menos adequado ao contexto de aplicação visado. O Shelly Pro 2PM, por sua vez, é um interruptor inteligente de duas entradas, vocacionado essencialmente para o controlo de estores e persianas. A

sua reduzida quantidade de entradas e a ausência de um ambiente de programação estruturado, baseado em linguagens normalizadas como as previstas pela norma IEC 61131-3, limitavam significativamente a implementação da lógica de controlo pretendida.

O controlador escolhido foi então o Arduino Opta, representado na Figura 3.4, que se trata de um micro-PLC industrial desenvolvido pela Arduino em parceria com a Finder, fabricante de referência em automação industrial e de edifícios. Consiste num autómato programável de fácil acessibilidade e foi concebido para operar em ambientes industriais exigentes, apresentando uma temperatura de operação entre -20°C e $+50^{\circ}\text{C}$, bem como grau de proteção IP20. A sua fiabilidade é reforçada pelas certificações CE, ENEC e cULus, que atestam a conformidade com os requisitos normativos europeus e norte-americanos para equipamentos de automação industrial.



Figura 3.4: Arduino Opta.

O autómato dispõe de oito entradas configuráveis individualmente como digitais ou analógicas, com uma gama de leitura compreendida entre 0 e 10 V para entradas analógicas e 0 a 24 V para digitais. Por outro lado, o equipamento incorpora quatro saídas de relé, que funcionam como interruptores eletromecânicos controlados por um sinal elétrico de baixa potência. Quando ativado pelo controlador, o relé fecha o circuito de saída, permitindo a passagem de corrente para o dispositivo externo. Quando desativado, o circuito é aberto e a passagem de corrente interrompida. Estas saídas têm uma capacidade de comutação até 250 VAC e 10 A por canal, correspondendo a uma potência máxima de 2.3 kW por relé. A comutação implica que as saídas sejam eletricamente isoladas do circuito de controlo, o que constitui uma vantagem em aplicações industriais onde é necessário separar os circuitos de sinalização dos circuitos de potência. O equipamento dispõe, ainda, de um botão programável pelo utilizador que pode ser configurado como entrada adicional, e um conjunto de LEDs também programáveis. O respetivo *Pinout* do autómato está representado na Figura 3.5.

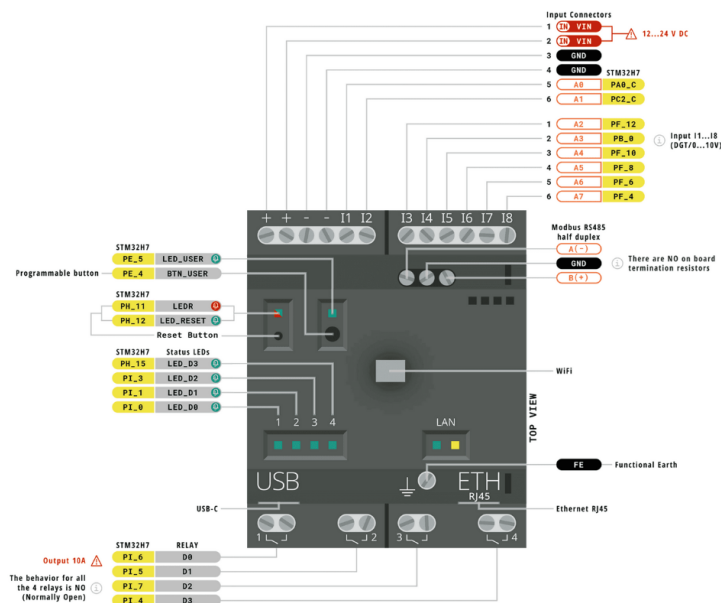


Figura 3.5: Pinout Opta.

O equipamento está disponível em três variantes, que diferem essencialmente ao nível das interfaces de comunicação, conforme sintetizado na Tabela 3.1. O modelo utilizado neste projeto é o Opta WiFi (AFX00002), que constitui a variante com maior capacidade de comunicação da família, incorporando Ethernet, RS-485, Wi-Fi e *Bluetooth Low Energy*.

Tabela 3.1: Versões do Arduino Opta.

	Opta WiFi	Opta RS485	Opta Lite
SKU	AFX00002	AFX00001	AFX00003
Ethernet	✓	✓	✓
USB-C	✓	✓	✓
RS485	✓	✓	
Wi-Fi	✓		
Bluetooth	✓		

A interface USB-C é utilizada para programação e depuração do equipamento, permitindo a ligação direta a um computador para transferência de código e monitorização do sistema em desenvolvimento, é a camada necessária para alterações na lógica criada.

A interface de Ethernet permite a ligação do equipamento a redes locais, suportando comunicação TCP/IP e o protocolo Modbus TCP, amplamente utilizado em sistemas de automação industrial e de edifícios. A interface RS-485, disponível nas variantes, opera em modo semi-duplex e suporta o protocolo Modbus RTU, permitindo a comunicação série com dispositivos industriais

como sensores, atuadores e outros controladores. A interface Wi-Fi, exclusiva do modelo utilizado neste projeto, opera na banda de 2.4 GHz, permitindo a ligação do equipamento a redes sem fios locais, possibilitando a comunicação com outros dispositivos e sistemas sem necessidade de cablagem dedicada, o que é particularmente relevante em instalações onde a infraestrutura de rede com fios é limitada ou inexistente, ou onde se pretende reduzir a complexidade da instalação. É de notar que o Opta não suporta redes Wi-Fi na banda de 5 GHz, o que constitui uma limitação a ter em conta na configuração da rede local. Esta limitação revelou-se, na prática, uma das dificuldades encontradas no decorrer do projeto, dado que a infraestrutura de rede da empresa dispunha de duas redes Wi-Fi distintas, uma na banda de 2.4 GHz e outra na banda de 5 GHz, sendo esta última a utilizada preferencialmente. Em situações em que apenas a rede de 5 GHz se encontrava ativa, o Opta ficava impossibilitado de estabelecer ligação, exigindo a disponibilização explícita da rede de 2.4 GHz para garantir a conectividade do equipamento.

Para o desenvolvimento da lógica de controlo associada ao *AirLight* foi usado o *software* Arduino PLC IDE, que é particularmente interessante pela sua capacidade de suportar dois ambientes de programação distintos e complementares dentro da mesma aplicação. Por um lado, suporta programação através de um *sketch*, utilizando a linguagem Arduino baseada em C++, com um vasto ecossistema de bibliotecas. Por outro, suporta programação em linguagens normalizadas pela norma IEC 61131-3, nomeadamente *Ladder Diagram* (LD) e *Structured Text* (ST). Estes dois ambientes podem coexistir no mesmo equipamento, permitindo que diferentes partes do sistema sejam desenvolvidas na linguagem mais adequada a cada função. O Opta incorpora ainda um relógio em tempo real (RTC), com retenção típica de 10 dias sem alimentação, que permite ao equipamento manter a noção do tempo mesmo em caso de falha de energia. A sincronização do RTC pode ser realizada via NTP (*Network Time Protocol*) através da interface Ethernet, garantindo a precisão temporal do sistema em aplicações que dependam de eventos ou registos baseados em tempo.

3.4 Alimentação, Sensores e Entrada

A interação do sistema com o ambiente em que opera e com o utilizador é assegurada por um conjunto de entradas físicas ligadas ao autómato, complementado por uma fonte de alimentação dedicada. As entradas correspondem a botões físicos para comando manual local e a sensores que detetam condições ambientais relevantes para o funcionamento do sistema, nomeadamente a temperatura e a ocorrência de precipitação.

A alimentação do sistema é assegurada por uma fonte do tipo NDR-240-24, uma fonte AC/DC industrial com tensão de saída de 24 VDC, projetada para montagem em calha DIN e apresentada na Figura 3.6. Esta tensão de saída cumpre o requisito de alimentação a 24 VDC dos atuadores, discutido na Secção 3.2. A escolha desta fonte justifica-se também pela sua adequação ao contexto em que o sistema se insere, uma vez que serve para alimentar simultaneamente o autómato e o atuador, o que dispensa a utilização de fontes auxiliares com diferentes níveis de tensão. Por outro lado o seu formato em calha DIN permite uma integração coerente com os restantes

equipamentos do quadro elétrico. No caso do autômato, a alimentação externa é essencial para o funcionamento das entradas e saídas industriais, garantindo a diferença de potencial necessária para que as entradas detetem mudanças de estado dos sensores e para que os relés de saída comutem. A fonte dispõe ainda de proteções integradas contra curto-circuito, sobrecarga, sobretensão e sobreaquecimento, garantindo a fiabilidade do fornecimento de energia ao sistema.

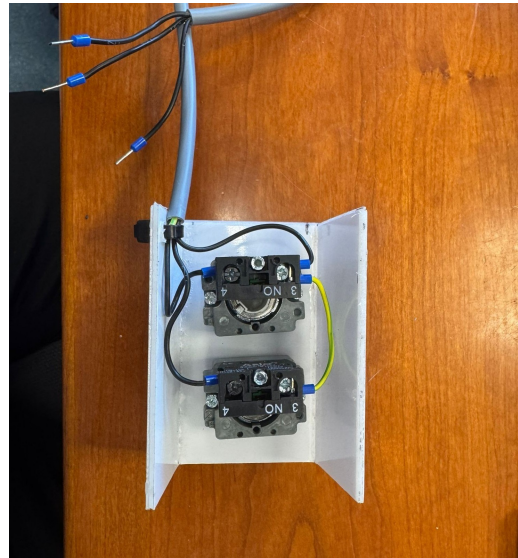


Figura 3.6: Fonte de alimentação.

O controlo manual do sistema é realizado através de um conjunto de duas botoneiras de pressão momentânea instaladas junto ao quadro do autômato, conforme ilustrado na Figura 3.7. Estes botões são equipados com contactos do tipo normalmente aberto (NO), permanecendo eletricamente abertos em repouso e fechando o circuito apenas durante o tempo em que são pressionados pelo utilizador. A ligação ao autômato é feita através de um terminal comum, alimentado a 24 VDC, partilhado por ambos os botões, e de dois cabos individuais ligados a entradas digitais distintas do Opta. Quando um botão é premido, o respetivo cabo é colocado a 24 VDC, gerando uma transição lógica na entrada correspondente do autômato.



(a) Vista 1.



(b) Vista 2.

Figura 3.7: Botoneiras para controlo manual.

A perceção das condições ambientais relevantes para o funcionamento do sistema é assegurada por dois sensores: um termóstato e um sensor de chuva. Ambos são dispositivos com saída digital do tipo contacto seco, que se ligam às entradas digitais do autómato. Quando a condição que detetam é verificada, fecham o circuito ligado à entrada do Opta, fazendo com que o nível lógico passe de 0 para 1 e desencadeando a leitura por parte do programa de controlo.

O termóstato é um dispositivo eletromecânico com *setpoint* de temperatura configurável manualmente. Quando a temperatura ambiente ultrapassa o valor definido, o seu contacto interno fecha, sinalizando ao autómato a ocorrência da condição. O sensor utilizado apresenta por defeito um *offset* de atraso de 2 graus, ou seja, se definirmos um *setpoint* de 24 °C, o contacto só vai fechar aos 26 °C. Este comportamento corresponde a uma histerese intencional, característica comum em termóstatos mecânicos deste tipo. Ao impor um desvio entre o ponto de fecho e o ponto de abertura do contacto, evita-se a comutação repetida do relé quando a temperatura ronda o valor definido, prolongando a vida útil do dispositivo e estabilizando o comportamento do sistema. Na Figura 3.8 está ilustrado o sensor de temperatura.



Figura 3.8: Sensor de temperatura.

O sensor de chuva utilizado possui uma superfície sensível à presença de água, do tipo PCB verde exposta com padrão concêntrico em ouro com duas pistas condutoras entrelaçadas formando um circuito aberto em seco, como demonstra a Figura 3.9. Quando a água cobre a superfície, cria uma ponte resistiva entre as pistas, alterando a condutância medida e sinaliza o autômato.

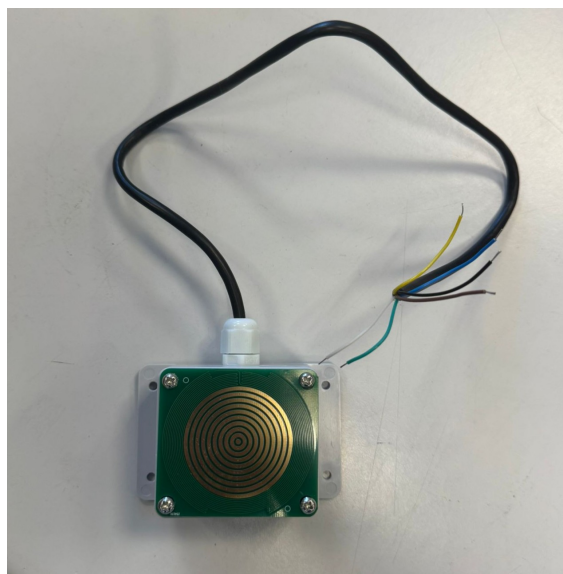


Figura 3.9: Sensor de chuva.

Ambos os sensores funcionam de forma autônoma, sem necessidade de configuração via software, e o autômato limita-se a ler o estado das suas saídas. Adicionalmente também é usada uma outra entrada do sistema para a integração com uma central de desenfumagem, que será abordado na secção devida.

3.5 Estágio de comutação

As saídas de relé do autômato funcionam como interruptores que abrem ou fecham um circuito, mas não aplicam diretamente tensão a sistemas externos como o atuador. Para permitir o movimento nos dois sentidos foi necessário introduzir um estágio intermédio de potência capaz de alimentar o motor com a polaridade correta em cada modo de operação. Este estágio é composto por dois relés montados em bases próprias, cujas bobinas são energizadas pelas saídas do Opta. Estes estão representados na Figura 3.10

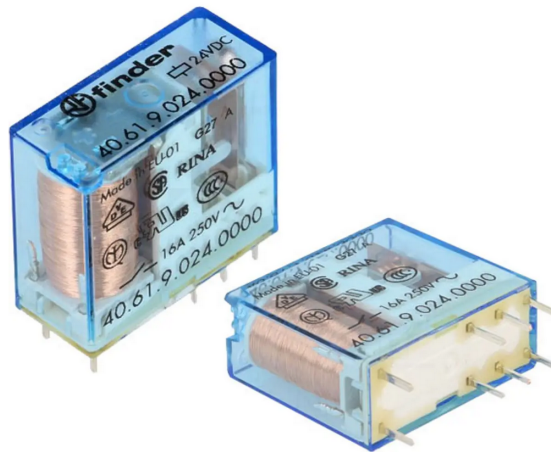


Figura 3.10: Relés do estágio de comutação.

Os relés utilizados são do tipo SPDT (*Single Pole Double Throw*). Em repouso, o terminal 11 (comum) está ligado ao terminal 12 através de um contacto normalmente fechado (NC). Quando a bobina é energizada, o estado do relé inverte-se: o contacto entre os terminais 11 e 12 abre, e o terminal comum passa a estar ligado ao terminal 14 através de um contacto normalmente aberto (NO).

A montagem usada liga o positivo da fonte aos terminais 14 de ambos os relés e o negativo aos terminais 12. Os terminais comuns de cada relé constituem os pontos de saída para o motor. Com ambos os relés em repouso, os dois terminais do motor ficam ligados ao negativo da fonte através dos contactos NC, anulando a diferença de potencial e mantendo o motor parado.

O comando de abertura ou fecho é efetuado ativando o relé correspondente. Ao ser energizado, o terminal comum desse relé passa a estar ligado ao positivo da fonte através do contacto NO, enquanto o terminal comum do outro relé se mantém ligado ao negativo. Esta configuração aplica ao motor uma diferença de potencial cuja polaridade depende de qual dos relés está ativo, determinando assim o sentido de rotação. Para parar o movimento, basta desativar o relé em atuação, restabelecendo a ligação ao negativo em ambos os terminais e anulando a diferença de potencial aplicada. A Figura 3.11 representa o esquemático dos relés.

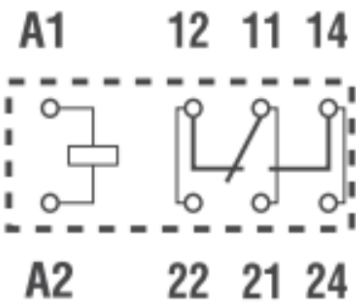


Figura 3.11: Esquema de funcionamento dos Relés.

3.6 Esquema de Ligações

A integração física dos componentes descritos nas secções anteriores foi realizada com montagem em calha DIN, conforme demonstra a Figura 3.13. Esta organização, permite manter um agrupamento coerente dos equipamentos e facilitar tanto a instalação como eventuais intervenções de manutenção. O sistema completo inclui, então, a fonte de alimentação, o autómato e os relés do estágio de comutação dispostos em calha DIN, bem como os sensores e a caixa de botoneiras instalados externamente. A Figura 3.12 apresenta o esquemático das ligações elétricas do sistema.

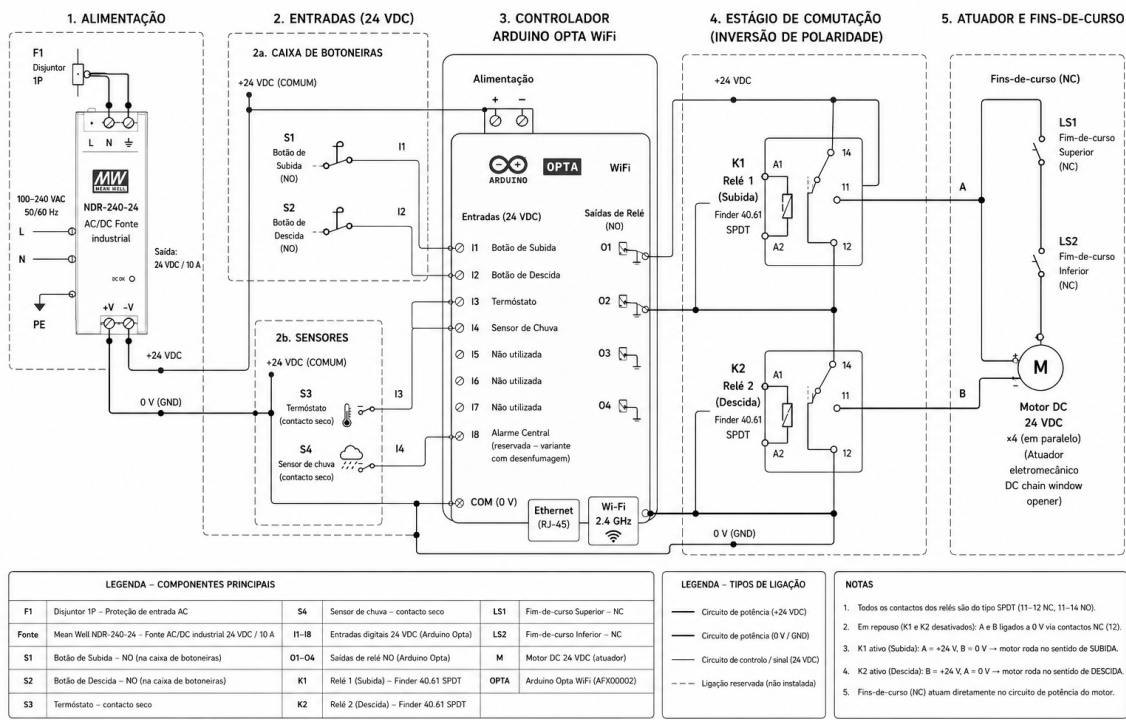


Figura 3.12: Esquemático das ligações elétricas do sistema de controlo.

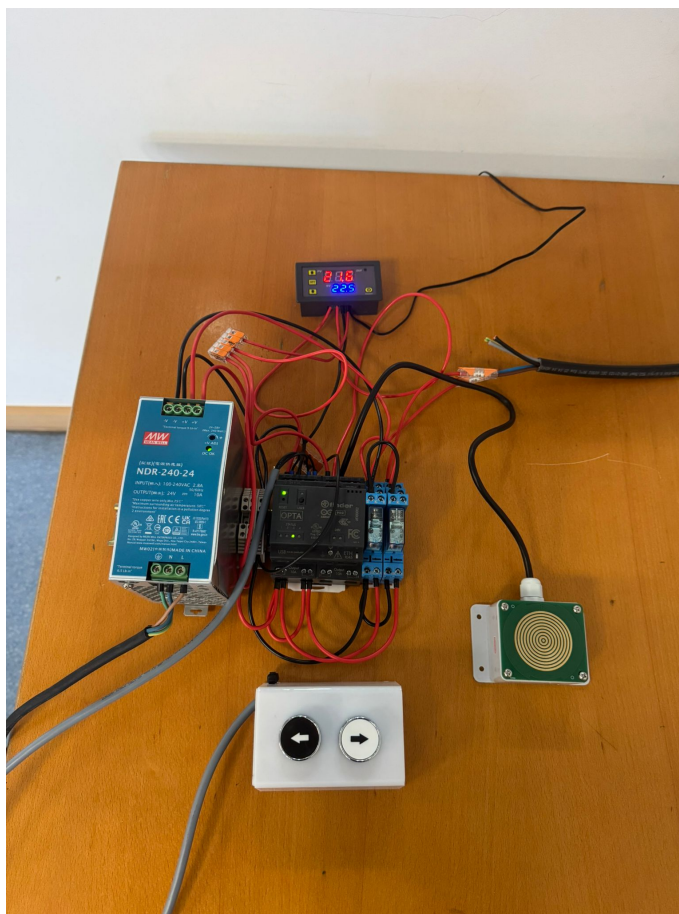


Figura 3.13: Disposição dos componentes em calha DIN e sensores.

3.7 Estudo de integração com uma central de desenfumagem

A empresa Chatron comercializa, para além do AirLight, soluções de desenfumagem que recorrem a centrais certificadas para o comando de atuadores em situações de emergência. A possibilidade de o AirLight vir a coexistir com este tipo de equipamento motivou o estudo descrito nesta secção, cujo objetivo foi avaliar a viabilidade técnica de integrar o sistema de controlo desenvolvido com uma central de desenfumagem certificada, mantendo a separação funcional entre conforto e segurança. Este estudo não constitui uma alteração ao sistema de controlo base do AirLight descrito nas secções anteriores, tratando-se de uma vertente exploratória conduzida em paralelo, com recurso a uma central Actulux SV 24V disponibilizada pela empresa para efeitos de teste.

A abordagem adotada parte de um princípio inviolável: a função de desenfumagem deve permanecer da responsabilidade exclusiva da central, não podendo o sistema de controlo desenvolvido ter qualquer tipo de influência nessa funcionalidade. Sob esta premissa, foi testada uma arquitetura em que o Opta deixa de comandar diretamente os atuadores e passa a comandar as entradas de conforto da central, sendo esta a responsável por alimentar os atuadores em qualquer situação.

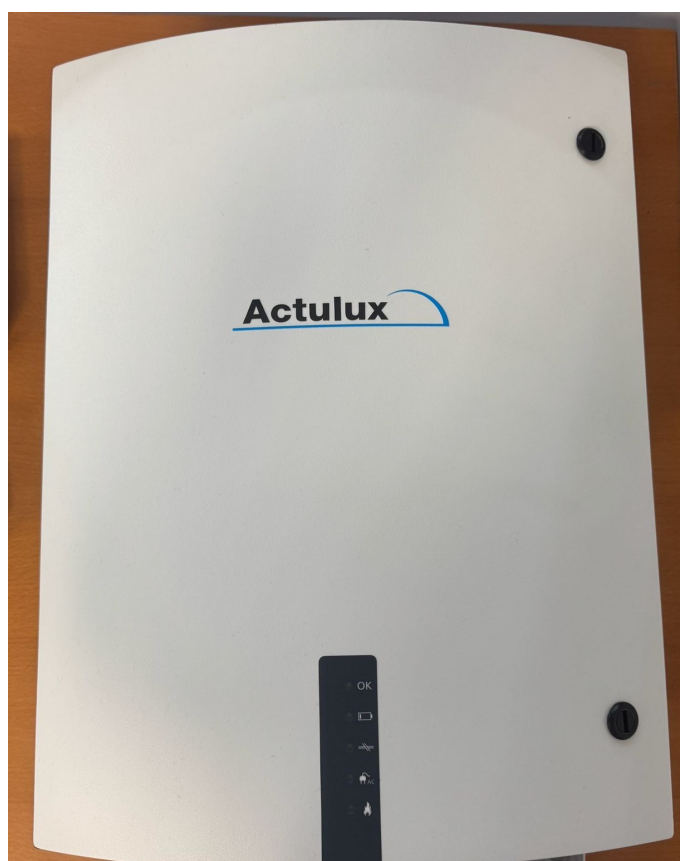


Figura 3.14: Central de Desenfumagem Actulux.

O comando da central em regime normal é realizado através das suas entradas de conforto. As duas saídas de relé do Opta são ligadas aos terminais *Up* e *Down* da entrada *Comfort 1* da central, comandando respectivamente a abertura e o fecho. A tensão de referência necessária a estas entradas é fornecida pelo Opta, sendo a sua saída de tensão ligada ao terminal comum da entrada *Comfort 1*. Desta forma, as saídas de relé comutam a tensão de referência para o terminal *Up* ou *Down* consoante o movimento pretendido, sendo a central a responsável por alimentar diretamente os atuadores.

A ligação entre a central e os atuadores requer um dispositivo de monitorização designado UniMon, representado na Figura 3.15 e recomendado pelo fabricante para os casos em que não são utilizados atuadores da própria Actulux. Os pinos 1 e 2 do UniMon são ligados, respetivamente, aos pinos 3 e 2 da zona *Actuator 1* da central, enquanto os terminais positivo e negativo do UniMon alimentam os atuadores. Ao contrário do sistema base, em que era necessário um estágio de comutação por relés para inverter a polaridade aplicada ao motor, as saídas de atuador da central realizam essa inversão de polaridade autonomamente. Por esse motivo, os relés deixam de ser necessários nesta configuração, sendo os quatro atuadores ligados em paralelo diretamente aos terminais positivo e negativo do UniMon e alimentados em conjunto a partir de uma única saída da central.



Figura 3.15: Dispositivo UniMon.

Para que o sistema de controlo tenha conhecimento do estado da central, a saída de alarme desta é ligada a uma entrada digital do Opta. A entrada I8 do controlador é ligada aos terminais *Com7* e *Com8* da saída *Alarm.Out* da central, permitindo ao Opta ler o estado de alarme sem nunca o comandar. Esta ligação garante que, em caso de alarme, o sistema de controlo tem conhecimento da situação e pode reagir em conformidade, mantendo a função de desenfumagem como responsabilidade exclusiva da central.

A ativação manual da função de desenfumagem é assegurada por um painel de incêndio da Actulux, ligado à central. Os terminais 13, 17A e 18 da central são ligados, respetivamente, aos terminais 4, 7 e 6 do painel. Em condições normais, uma resistência de fim-de-linha liga os terminais 13 e 17 da central, mantendo o circuito de supervisão monitorizado e impedindo a ativação indevida da função de incêndio. Aquando da instalação do painel, esta resistência é retirada, passando o painel a integrar o circuito de supervisão e a permitir o acionamento manual da desenfumagem. A Figura 3.16 ilustra o painel de incêndio.



(a) Interior.



(b) Exterior.

Figura 3.16: Painel de Incêndio.

A central dispõe de treze *DIP switches* que definem o seu comportamento, dos quais três são particularmente relevantes para a configuração adotada, representada na Figura 3.17. O *DIP 2 (Puls/Const)* é o mais crítico: tem de estar na posição *OFF (Const)*, para que o atuador se mova enquanto o sinal de comando estiver presente. Na posição *ON (Puls)*, a central limitaria cada comando de abertura a um pulso fixo de seis segundos, inviabilizando as funcionalidades pretendidas neste projeto. O *DIP 7 (Gr. 1+2 comfort)* é colocado em *OFF*, fazendo com que a entrada *Comfort 1* comande apenas a saída *Actuator 1*, coerentemente com a opção de ligar os quatro atuadores em paralelo numa única saída. Por fim, o *DIP 4 (Firesw. 2 ch)* é colocado em *OFF*, configurando a central com um único grupo de incêndio, de modo a que, em situação de alarme, a totalidade dos atuadores abra simultaneamente.

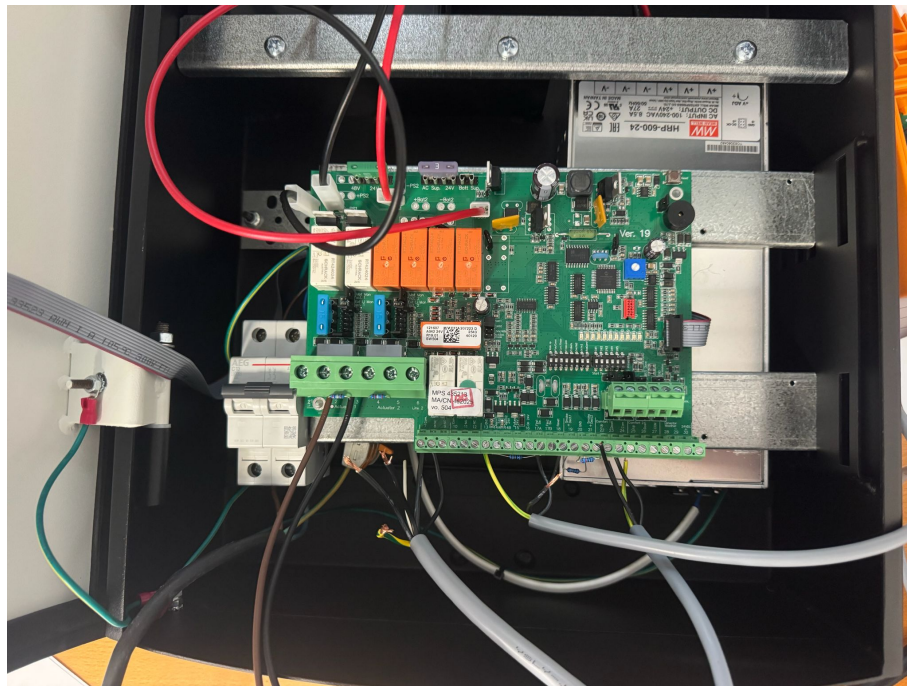


Figura 3.17: Configuração da Central de Desenfumagem.

A integração com a central de desenfumagem altera de forma significativa a arquitetura física do sistema base. O comando direto dos atuadores pelo estágio de comutação deixa de ocorrer, passando o controlador a comandar a central através das suas entradas de conforto, e sendo a central a alimentar os atuadores através do UniMon, conforme representado na Figura 3.18. O sistema passa, ainda, a dispor de um sinal de alarme lido pelo OPTA e de um painel manual de incêndio ligado à central. O resultado é um sistema em que a função de conforto permanece sob o comando do OPTA, mas em que a função de segurança fica integralmente assegurada pela central certificada sob a norma EN12101, satisfazendo o princípio de separação entre as duas funções.

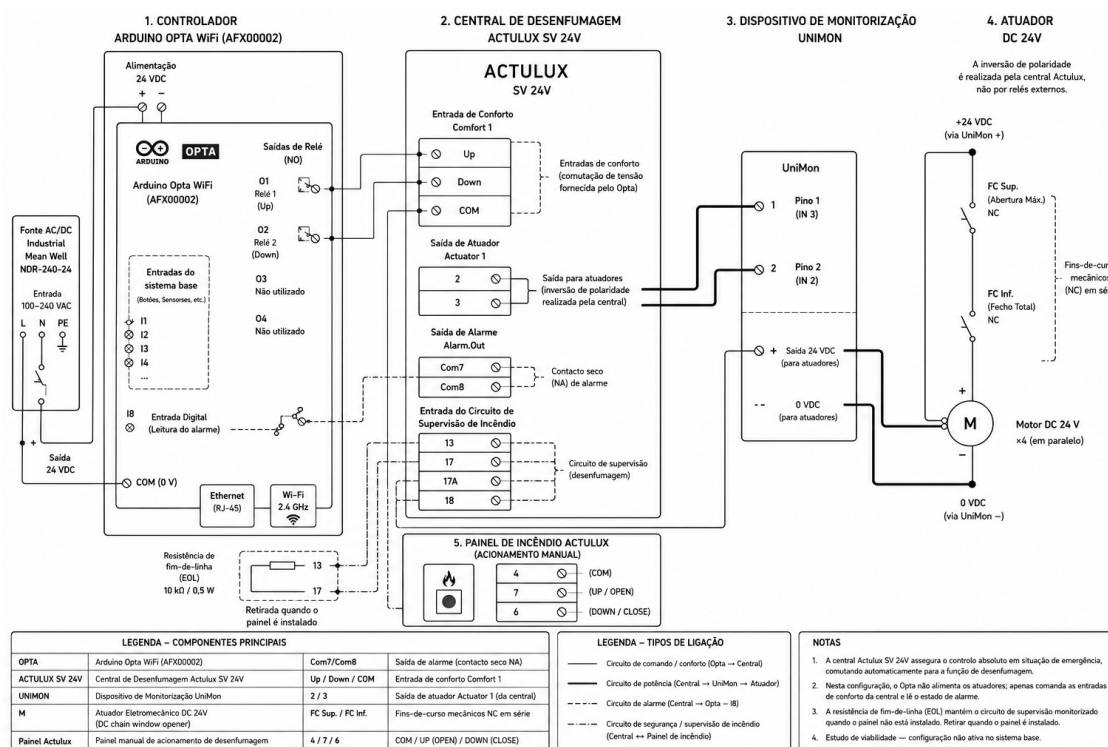


Figura 3.18: Esquemático das ligações elétricas na configuração estudada com central de desenfumagem.

3.8 Síntese do sistema

O sistema de controlo apresentado neste capítulo assenta então numa arquitetura responsável por adquirir informação proveniente dos sensores ambientais e dos comandos do utilizador, e por comandar o movimento do *AirLight* através de um estágio de comutação. A proteção mecânica do equipamento é assegurada por fins-de-curso que atuam diretamente sobre o circuito de potência, independentemente do controlador, constituindo uma camada de segurança autónoma. A integração da fonte de alimentação, do controlador, dos relés, dos sensores e das botoneiras numa calha DIN resulta numa solução compacta e adequada ao contexto industrial.

O estudo de integração com a central de desenfumagem Actulux demonstrou igualmente a viabilidade de uma arquitetura alternativa, na qual o controlador atua apenas sobre as entradas de conforto da central, ficando o acionamento direto dos atuadores a cargo desta. Esta abordagem preserva a separação entre as funções de conforto e de segurança contra incêndio, permitindo simultaneamente adaptar o sistema a aplicações com requisitos de certificação mais exigentes. Embora esta configuração não faça parte do sistema base desenvolvido, a sua análise evidencia a flexibilidade e a capacidade de adaptação da arquitetura proposta.

O capítulo seguinte apresenta o desenvolvimento da lógica de controlo implementada no Arduino Opta, bem como das camadas de comunicação e supervisão que complementam a arquitetura de hardware descrita neste capítulo.

Capítulo 4

Desenvolvimento e Integração do Sistema

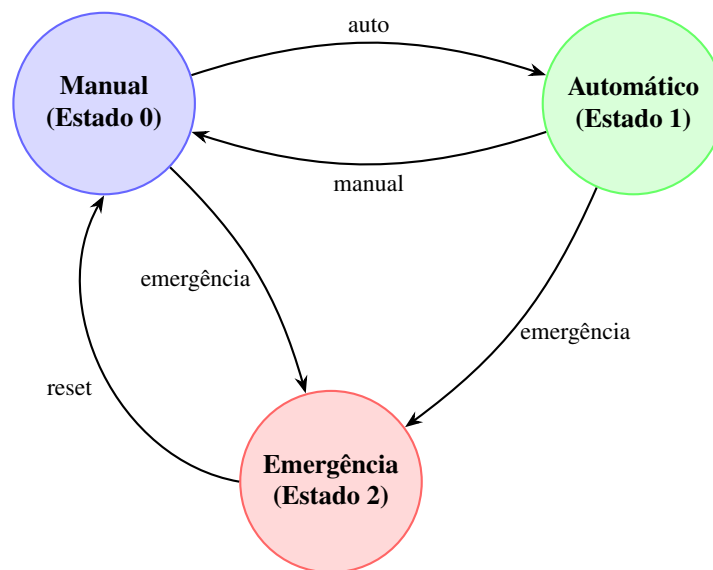
O presente capítulo aborda o desenvolvimento da lógica de controlo e das camadas de *software* que conferem funcionalidade ao sistema. Parte-se da lógica implementada no autómato e prossegue-se para os mecanismos que permitem a interação do sistema com plataformas externas. Aborda-se também a integração do sistema de controlo com a central de desenfumagem e o capítulo encerra com a descrição do fluxo de controlo completo, que integra todos estes elementos.

4.1 Arquitetura da lógica de controlo

A lógica de controlo implementada no autómato, através do *software* Arduino PLC IDE, constitui o núcleo funcional do sistema, sendo responsável por interpretar os comandos recebidos, quer das interfaces de utilizador ou das entradas físicas, e traduzi-los em ações sobre o atuador, respeitando as condições de segurança e as prioridades definidas. Esta lógica gere os diferentes modos de operação do sistema, controla o movimento de abertura e fecho do equipamento, e assegura que diferentes situações e ordens são tratadas de forma coerente e previsível. O sistema de controlo está organizado em três programas que isolam a decisão da execução: *ModeMachine*, *AutoControl* e *Relay Control*.

O *ModeMachine* ocupa o nível mais alto, gerindo os modos de operação do sistema e decidindo as ações tomadas, de acordo com o modo ativo e as prioridades definidas.

A escolha do modo de operação ativo, baseia-se numa máquina de estados, ilustrada na Figura 4.1 e composta por três estados principais: Manual, Automático e Emergência. A gestão da transição entre os diferentes modos é assegurada através de comandos provenientes do utilizador, que podem surgir localmente ou remotamente, e também de condições de segurança.

Figura 4.1: Máquina de estados *ModeMachine*.

O modo manual é definido como o estado principal do sistema, onde o controlo do movimento é realizado diretamente pelo utilizador através dos botões físicos ou dos comandos externos. Neste modo, o operador tem controlo total sobre os movimentos de subida e descida do sistema, desde que sejam respeitadas as restrições impostas pelos limites mecânicos e pelas condições ambientais.

Quando é detetada chuva, o sistema força automaticamente o movimento de descida, independentemente dos comandos introduzidos pelo utilizador, garantindo assim a proteção do equipamento e a hierarquia de prioridades do sistema. Caso não exista precipitação, os movimentos são executados apenas quando solicitados pelo operador, desde que o sistema não se encontre já na posição limite correspondente, seja de abertura máxima ou mínima. A partir deste estado é possível efetuar a transição para o modo automático, através do botão *User* ou mediante a seleção do respetivo comando de operação. A ativação do modo de emergência pode também ser obtida através de um comando externo ou através do botão de emergência.

O modo automático por outro lado, permite que o sistema opere de forma autónoma com base em referências de posição e em informações provenientes dos sensores instalados. A deteção de chuva possui, também neste modo, prioridade sobre todas as restantes funções automáticas, dado que sempre que esta condição ocorre, qualquer movimento ou posicionamento automático em curso é interrompido e o sistema executa o movimento de descida até atingir a posição de fecho. Adicionalmente, o sistema integra uma funcionalidade associada ao sensor de temperatura, em que quando a condição térmica é ativada, é registada a posição atual do sistema e inicia-se um movimento automático de subida completo. Quando essa condição deixa de estar presente, o sistema pode regressar automaticamente à posição anteriormente registada, restaurando a configuração existente antes da ativação do sensor.

O utilizador pode abandonar o modo automático e regressar ao modo manual através do respetivo comando de seleção. Da mesma forma, a ativação de uma condição de emergência provoca a transição imediata para o estado Emergência.

O modo de emergência corresponde ao modo de segurança do sistema. Quando este estado é ativado, todos os comandos de movimento são imediatamente anulados, impedindo qualquer deslocação do equipamento. Durante este estado é ativada uma sinalização visual específica, baseada no funcionamento intermitente dos indicadores luminosos, de forma a informar o operador de que o sistema se encontra em condição de emergência. A recuperação deste estado não é efetuada automaticamente. O sistema permanece em emergência até que seja executada uma ação explícita do operador para regressar ao modo manual, garantindo assim que a situação é previamente reconhecida e validada pelo utilizador.

A lógica de controlo implementada estabelece uma hierarquia de prioridades entre os diferentes eventos e funcionalidades do sistema. As condições de emergência possuem a prioridade máxima, seguidas pelas condições de proteção associadas à deteção de chuva. Em seguida encontram-se as funções automáticas de posicionamento e de atuação térmica. Por último, são considerados os comandos normais de operação. A escolha desta abordagem garante que os requisitos de segurança e proteção do equipamento prevalecem sempre sobre os restantes modos de funcionamento.

Implementado em *Ladder Diagram* e representado na Figura 4.2, o *RelayControl*, consiste no resultado final do controlo desenvolvido, sendo responsável por traduzir os comandos *CmdSubir* e *CmdDescer* na atuação das saídas físicas dos relés, impedindo que ambos sejam acionados em simultâneo. O principal objetivo da lógica desenvolvida é, então, assegurar que estes comandos são ativados devidamente. A condição lógica implementada é $\text{RelaySubir} := \text{CmdSubir} \text{ AND NOT } \text{CmdDescer}$ e $\text{RelayDescer} := \text{CmdDescer} \text{ AND NOT } \text{CmdSubir}$.

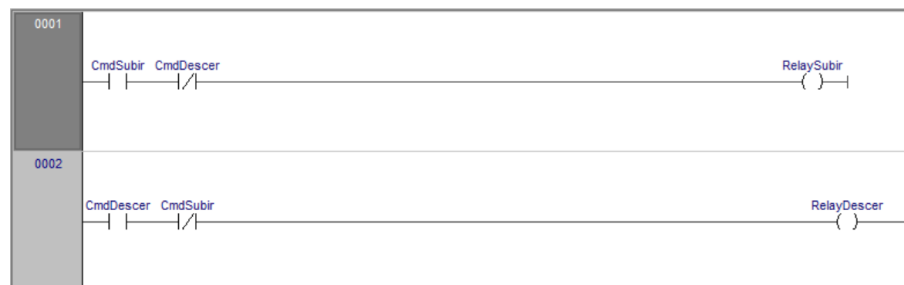


Figura 4.2: Programa *RelayControl*.

Esta proteção por *software* constitui uma camada adicional ao intertravamento elétrico já assegurado pelos contactos normalmente fechados dos relés, garantindo que os dois comandos opostos nunca são aplicados em simultâneo, mesmo perante um eventual erro na lógica de decisão.

O *AutoControl* traduz essas decisões em movimento, gerindo o estado físico do equipamento e mantendo uma estimativa da sua posição atual. O *RelayControl* fecha esta cadeia, sendo o princípio base deste controlo, por converter os comandos de movimento nas saídas físicas que acionam os relés. Neste programa é implementado o controlo de movimento do sistema, baseando-se também numa máquina de estados, como ilustra a Figura 4.3 e, tendo como principal objetivo averiguar em que estado o sistema se encontra, seja este um estado de repouso ou de movimento. As transições entre os diferentes estados estão condicionadas pelas próprias ações do sistema. Esta

lógica permite um maior controlo do sistema, dado que assim são impostos limites ao movimento e possibilita a integração da funcionalidade do controlo de posição.

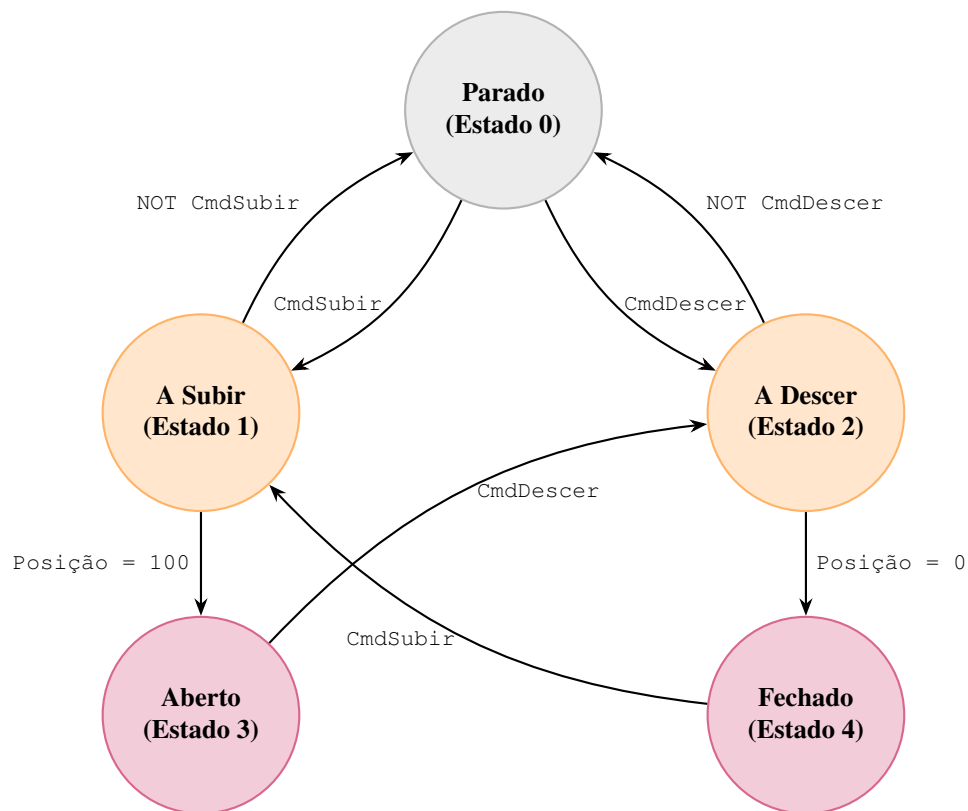


Figura 4.3: Máquina de estados *AutoControl*.

4.2 Controlo de Posição

Conforme discutido no capítulo 2, a utilização de posições intermédias revela-se uma estratégia mais eficaz que o controlo binário em sistemas de ventilação natural. A capacidade de posicionar o *AirLight* em qualquer ponto entre totalmente aberto e totalmente fechado constitui, por isso, um dos requisitos centrais do sistema desenvolvido. Para que o sistema seja capaz de atingir uma posição arbitrária, é necessário que o controlo registre a posição do equipamento em cada instante, dado que, sem essa informação, não é possível chegar a uma posição pretendida, nem decidir em que sentido se deve mover o atuador.

4.2.1 Método utilizado para o controlo de posição

A abordagem adotada para conseguir controlar a posição do sistema, baseia-se numa estimativa temporal em malha aberta, sem recurso a sensores físicos de posição, em que a posição estimada é atualizada a cada ciclo de execução do PLC, enquanto o atuador está num estado de movimento.

A viabilidade desta abordagem está diretamente dependente do conhecimento preciso do tempo total de curso do atuador em cada sentido. Para o efeito, foram conduzidos ensaios de caracterização cujos resultados são apresentados no Capítulo 5. Foram, então, assumidos como tempos médios, 38,00 s para abertura e 32,83 s para fecho, permitindo assim, calcular os incrementos a aplicar à estimativa de posição.

A posição é mantida numa variável chamada `CurrentPos`, com valores compreendidos entre 0,0 e 100,0, correspondendo respetivamente às posições totalmente fechada e totalmente aberta. Esta variável é incrementada enquanto o sistema se encontra no estado de subida e decrementada enquanto se encontra no estado de descida. O PLC executa o mesmo programa múltiplas vezes a cada segundo, sendo que cada execução é feita com intervalos de 10ms. Desta forma entre cada ciclo de execução nos estados de movimento é necessário atualizar a percentagem de posição com base num valor específico. Ao utilizarmos os tempos médios registados nos ensaios como referência, é possível calcular os incrementos a aplicar à estimativa de posição por cada ciclo, conforme as Equações 4.1 e 4.2.

$$\Delta_{\text{subida}} = \frac{100}{t_{\text{abertura}}/T_{\text{ciclo}}} = 0,025 \quad (4.1)$$

$$\Delta_{\text{descida}} = \frac{100}{t_{\text{fecho}}/T_{\text{ciclo}}} = 0,0312 \quad (4.2)$$

Os limites de posição são protegidos por duas camadas complementares, para além dos fins-de-curso, é usada também a lógica, implementada no PLC. A variável `CurrentPos` é mantida entre 0 e 100, com qualquer outro valor fora desta gama a ser rejeitado pela lógica de controlo. As duas camadas atuam em níveis diferentes e oferecem redundância no caso de uma delas falhar.

4.2.2 Funcionalidade de movimento por percentagem

Com a estimativa de posição mantida em tempo real e validada pela calibração descrita, torna-se possível implementar a funcionalidade de movimento por percentagem, que constitui a forma mais flexível de controlar o *AirLight*. Esta funcionalidade está disponível apenas em modo automático, opção que decorre de uma separação intencional de responsabilidades entre os dois modos de operação: o modo manual destina-se a um controlo direto e imediato do equipamento, através dos botões físicos ou da aplicação móvel, sem recurso a posicionamento calculado, enquanto o modo automático é o contexto em que a lógica de posicionamento tem aplicação. Esta funcionalidade permite definir uma posição-alvo entre 0% e 100%, seja por comandos do utilizador através da aplicação móvel, seja por regras definidas por sistemas de supervisão. O sistema é então responsável por mover o equipamento até essa posição alvo.

O comando é constituído por dois sinais: `targetPos`, que transporta o valor numérico da posição-alvo, e `mSetPos`, que sinaliza ao PLC a intenção de executar o movimento. A variável `mSetPos` funciona como gatilho, é enviada como pulso pelo cliente (passa a 1 e logo a 0) e o seu efeito é ativar a *flag* interna `SetPosActive`, que permanece verdadeira até que o movimento

esteja concluído. Esta separação entre o gatilho e a *flag* de estado permite que o ciclo de execução do PLC mantenha o controlo do movimento de forma autónoma após o pedido inicial.

Enquanto *SetPosActive* é verdadeira, a máquina de estados compara, em cada ciclo, o valor de *targetPos* com o valor atual de *CurrentPos*, desencadeando movimento de subida se a posição-alvo for superior, ou de descida se for inferior. O movimento prossegue até que a diferença entre as duas posições caia abaixo de uma tolerância de 0,05, momento em que *SetPosActive* é desativada e o movimento cessa. Esta tolerância corresponde a uma fração desprezável do curso total de movimento, equivalente a menos de meio milímetro num curso de várias dezenas de centímetros, sendo muito inferior à precisão prática exigida pela aplicação. A sua adoção tem, ainda, a vantagem de evitar oscilações decorrentes da representação dos valores em vírgula flutuante. O Algoritmo 1 sintetiza esta lógica, tendo ainda a vantagem de evitar oscilações decorrentes da representação dos valores em vírgula flutuante.

Algorithm 1 Movimento por percentagem

Require: *mSetPos*, *targetPos*

```

1: Estado: SetPosActive, CurrentPos

2: if mSetPos then
3:   if targetPos representa um comando auxiliar then
4:     executar comando auxiliar
5:   else if  $\neg SetPosActive$  then
6:     SetPosActive  $\leftarrow$  true
7:   end if
8: end if

9: if SetPosActive then
10:  if targetPos > CurrentPos + 0,05 then
11:    comandar movimento de subida
12:  else if targetPos < CurrentPos - 0,05 then
13:    comandar movimento de descida
14:  else
15:    parar movimento
16:    SetPosActive  $\leftarrow$  false
17:  end if
18: end if
  
```

Além dos valores válidos de posição, a variável *targetPos* é também utilizada para transportar comandos auxiliares ao algoritmo, nomeadamente para cancelar um movimento em curso ou para bloquear temporariamente a resposta automática ao sensor de temperatura. O valor -1,0 sinaliza ao PLC que deve cancelar um movimento de posicionamento em curso, deixando o sistema parado na posição atual, enquanto que o valor -2,0 sinaliza que a resposta automática ao sensor de temperatura deve ser bloqueada, suspendendo temporariamente a reação automática deste mesmo. Este mecanismo, e a sua articulação com o limite de variáveis impostas pelo *framework*, são detalhados na secção sobre a comunicação entre o sketch e o runtime do PLC.

4.2.3 Inicialização do sistema

Na implementação inicial do controlador, era assumido que o sistema se encontrava sempre na posição fechada ($CurrentPos = 0, 0$) no arranque do PLC. Esta hipótese é válida na maioria dos cenários, dado que o estado de repouso típico do *AirLight* é fechado, mas falha sempre que o PLC é reiniciado com o equipamento em posição aberta. Nesse cenário, o sistema assumiria erroneamente que se encontrava fechado, e todas as estimativas posteriores seriam corrompidas até uma calibração explícita. Para resolver esta limitação, foi implementada uma rotina de inicialização que aguarda confirmação do operador antes de entrar na lógica de controlo normal. No arranque, se o operador, observando fisicamente o sistema, pressionar o comando de descida, o sistema assume que o equipamento se encontra fechado e atribui $CurrentPos = 0, 0$, porém, se pressionar o comando de subida, assume que se encontra aberto e atribui $CurrentPos = 100, 0$.

Esta solução tira partido de uma característica física do sistema. Os fins-de-curso do *AirLight* cortam diretamente a corrente aos atuadores quando estes atingem os batentes extremos, sem mediação do controlador. Assim, mesmo que o operador pressione o comando de movimento com o sistema já na posição máxima desse sentido, não há movimento real, porque o fim-de-curso correspondente impede a alimentação dos atuadores. O comando é interpretado apenas pela lógica do PLC, para efeitos de inicialização da estimativa, sem produzir movimento físico indesejado. O Algoritmo 2 apresenta esta rotina.

Algorithm 2 Rotina de inicialização do sistema

Require: *CmdSubir, CmdDescer*

```

1: Estado: InitDone, AutoState, CurrentPos

2: if  $\neg InitDone$  then
3:   if CmdDescer then
4:     AutoState  $\leftarrow$  FECHADO
5:     CurrentPos  $\leftarrow$  0,0
6:     InitDone  $\leftarrow$  true
7:   else if CmdSubir then
8:     AutoState  $\leftarrow$  ABERTO
9:     CurrentPos  $\leftarrow$  100,0
10:    InitDone  $\leftarrow$  true
11:   end if
12:   return
13: end if

```

A instrução `RETURN` no final do bloco é essencial, por impedir que a restante lógica do programa *AutoControl* execute enquanto a inicialização não estiver concluída, garantindo que a máquina de estados não opera sobre valores indefinidos.

Importa reconhecer que a correção da inicialização depende de uma observação correta do operador. Uma seleção errada, como pressionar o comando de descida com o sistema fisicamente

aberto, ou o contrário, leva a uma estimativa inicial incorreta, corrompendo silenciosamente todas as decisões de posicionamento subsequentes até à calibração seguinte.

4.2.4 Calibração da estimativa

No decorrer do funcionamento normal do sistema, ao longo de múltiplos movimentos sucessivos, pequenos desvios entre o tempo teórico assumido e o tempo real de movimento, decorrentes de variabilidade, podem acumular-se, traduzindo-se num desfasamento progressivo entre a estimativa interna e a posição física real.

Para mitigar esta deriva, foi implementada uma rotina de calibração que recupera a coerência da estimativa recorrendo aos fins-de-curso físicos como referência absoluta. O princípio é direto: comandar uma descida durante um tempo deliberadamente superior ao tempo de fecho do sistema, garantindo que o fim-de-curso inferior é atingido independentemente da posição real de partida. O tempo de calibração escolhido foi de 35 s, sendo que quando este tempo se esgota, o sistema encontra-se com certeza na posição fechada e a estimativa é reposta a `CurrentPos = 0,0`.

Para ativar esta rotina, podemos fazê-lo de forma manual através do comando `mCalibrate` emitido pelo operador, ou de forma automática, com base num contador de movimentos, em que, sempre que o sistema completa um número predefinido de movimentos (quarenta, na implementação atual), uma *flag* interna assinala a necessidade de calibração. Na próxima vez que o sistema chegue naturalmente à posição assumida como 0%, a rotina é desencadeada de forma oportunista, prolongando o fecho durante o tempo de calibração. Esta abordagem evita a interrupção de operações em curso, visto que o sistema não é forçado a fechar arbitrariamente, mas apenas aproveita um fecho que já estava a acontecer.

A contagem de movimentos baseia-se na máquina de estados, *AutoControl*, sendo que esta abordagem assegura que tanto os movimentos por comando manual como os movimentos por posicionamento percentual são contabilizados de forma uniforme. Quando a calibração é executada, por qualquer das vias, o contador é reposto a zero, articulando assim as duas formas de calibração, uma calibração manual executada antes do limiar reinicia o ciclo de contagem, dispensando uma calibração automática iminente.

A máquina de estados do programa *AutoControl*, no seu funcionamento normal, não permite o comando de descida quando o sistema se encontra no estado Fechado. Este comportamento, embora correto na operação regular, impediria a calibração de funcionar, pois se a estimativa interna acreditar que o sistema já está em `CurrentPos = 0,0`, a máquina de estados nunca permitiria descer mais. Para contornar esta limitação, a rotina de calibração executa antes da máquina de estados e atribui o comando de descida diretamente, terminando o bloco com `RETURN`. A máquina de estados não chega a ser executada durante a calibração, pelo que o filtro normal não se aplica e o comando é mantido até o tempo esgotar. O Algoritmo 3 apresenta a rotina completa, incluindo a contagem de movimentos e os dois gatilhos de ativação.

Algorithm 3 Rotina de calibração automática e manual**Require:** *mCalibrate*, *AutoState*

```

1: Estado: MoveCount, CalibNeeded, CalibActive, PrevState

2: if (PrevState = 1 ou PrevState = 2) e AutoState ≠ 1 e AutoState ≠ 2 then
3:   MoveCount ← MoveCount + 1
4:   if MoveCount ≥ 40 then
5:     CalibNeeded ← true
6:   end if
7: end if
8: PrevState ← AutoState

9: if mCalibrate then
10:  CalibActive ← true
11: end if
12: if CalibNeeded e AutoState = FECHADO e ¬CalibActive then
13:  CalibActive ← true
14: end if

15: if CalibActive then
16:  CmdSubir ← false
17:  CmdDescer ← true
18:  if tempo de calibração esgotado then
19:    CmdDescer ← false
20:    CurrentPos ← 0,0
21:    AutoState ← FECHADO
22:    CalibActive ← false
23:    CalibNeeded ← false
24:    MoveCount ← 0
25:  end if
26:  return
27: end if

```

4.3 Servidor HTTP e interface de comunicação

A lógica de controlo desenvolvida no autómato seria, por si só, insuficiente para o caso de uso pretendido se o sistema estivesse isolado de comunicação com sistemas externos. A capacidade de comandar o *AirLight* remotamente, de consultar o seu estado e de o integrar com camadas de supervisão são funcionalidades necessárias para o sistema, estando dependentes da existência de uma arquitetura de comunicação que permita a interação com clientes externos.

4.3.1 Variáveis partilhadas entre *sketch* e PLC

A comunicação do Opta com sistemas externos pode ser assegurada pela interface Wi-Fi do equipamento, configurada diretamente no Arduino PLC IDE e representada na Figura 4.4. Neste bloco são definidos o SSID e a palavra-passe da rede, bem como a configuração de rede estática:

endereço IP, máscara de sub-rede, *gateway* e servidor DNS. A utilização de um endereço IP estático garante que o equipamento mantém um endereço previsível ao longo do tempo, permitindo que os clientes comuniquem com ele de forma fiável sem mecanismos de descoberta.

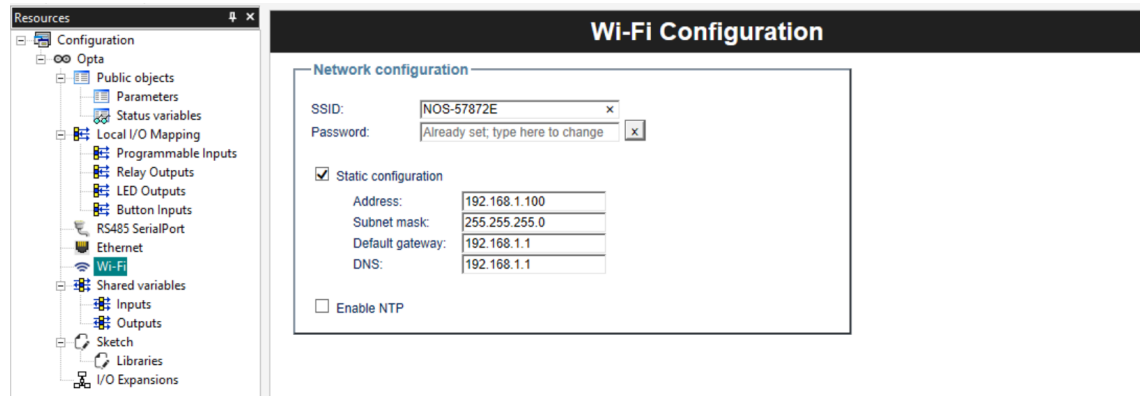
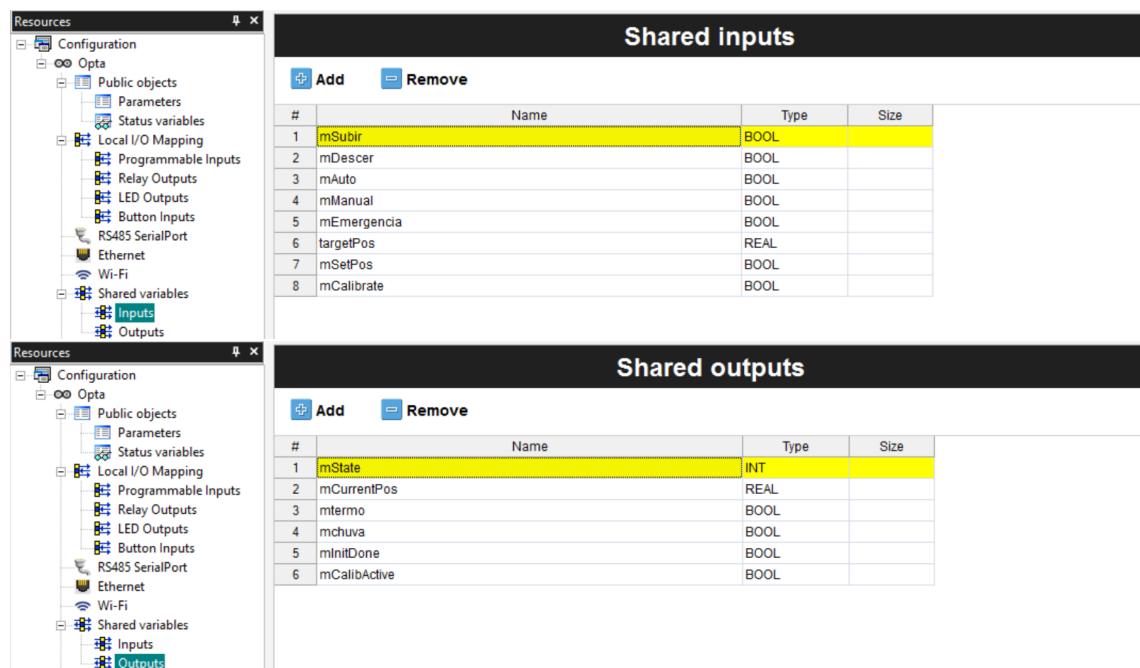


Figura 4.4: Configuração Wi-Fi no Arduino PLC IDE.

Quando o Opta é alimentado, o *runtime* estabelece automaticamente a ligação à rede Wi-Fi com os parâmetros configurados. A partir desse instante, o equipamento fica acessível na rede local, sendo a sua presença verificável através de um comando `ping` ao endereço configurado.

Conforme descrito no Capítulo 3, o autómato permite a execução simultânea de um *sketch* Arduino e de programas PLC no mesmo equipamento. A troca de informação entre os dois ambientes é assegurada por um conjunto de variáveis partilhadas designadas *Shared Variables*, declaradas no Arduino PLC IDE e acessíveis tanto ao *sketch* como aos programas PLC.

Internamente, este mecanismo é implementado pelo *framework* recorrendo a registos *Modbus* em memória partilhada, ainda que, do ponto de vista do programador, a abstração apresentada seja a de variáveis nativas em ambos os ambientes. As variáveis dividem-se em duas categorias: *Shared Inputs*, escritas pelo *sketch* e lidas pelos programas PLC, e *Shared Outputs*, escritas pelos programas PLC e lidas pelo *sketch*. As *Shared Variables* usadas na lógica estão apresentadas na Figura 4.5.



#	Name	Type	Size
1	mSubir	BOOL	
2	mDescer	BOOL	
3	mAuto	BOOL	
4	mManual	BOOL	
5	mEmergencia	BOOL	
6	targetPos	REAL	
7	mSetPos	BOOL	
8	mCalibrate	BOOL	

#	Name	Type	Size
1	mState	INT	
2	mCurrentPos	REAL	
3	mtermo	BOOL	
4	mchuva	BOOL	
5	mInitDone	BOOL	
6	mCalibActive	BOOL	

Figura 4.5: Shared Variables.

Uma restrição imposta pelo *framework* do Arduino PLC IDE é o limite de oito *Shared Inputs* declaráveis em simultâneo. Esta limitação condicionou o desenho da interface e impossibilitou a criação de variáveis dedicadas para funcionalidades adicionais que, à medida que o sistema evoluiu, foi necessário acomodar. Para contornar esta restrição, recorreu-se à multiplexação semântica da variável *targetPos*, atribuindo valores fora do intervalo válido de posicionamento (0 a 100) a comandos qualitativos distintos, tal como descrito na secção anterior.

4.3.2 Endpoints REST

Para expor as variáveis partilhadas ao exterior, o *sketch* implementa um servidor HTTP na porta 80, que constitui a interface de comunicação externa do Opta. É constituído por duas funções cada uma com objetivos diferentes: `setup()`, responsável por instanciar o servidor e coloca-o à escuta, enquanto a função `loop()` processa os pedidos recebidos e constrói as respostas, sendo executada continuamente. O servidor funciona como ponte entre o exterior e a lógica de controlo, sendo capaz de traduzir pedidos HTTP em escritas às *Shared Inputs* e devolver aos clientes, em formato JSON, os valores das *Shared Outputs*.

A implementação do servidor HTTP no *sketch* segue uma abordagem minimalista, adequada ao contexto do projeto, dado que é usado um equipamento com recursos de memória e processamento limitados. Em vez de recorrer a bibliotecas HTTP de alto nível, o *sketch* implementa um *parser* manual que extrai o método, o caminho e o corpo de cada pedido com base na estrutura HTTP fundamental. É feita a leitura dos cabeçalhos até ao delimitador de fim de cabeçalho, a extração do tamanho do corpo a partir do cabeçalho `Content-Length`, bem como a leitura do

número exato de *bytes* indicado. Cada resposta inclui o cabeçalho `Connection: close`, encerrando a ligação após cada troca e o servidor processa um único pedido de cada vez, ou seja, enquanto um cliente estiver a ser atendido, pedidos adicionais aguardam.

A interface do *sketch* é constituída por dois *endpoints*. O `GET /api/status` devolve uma resposta JSON com o estado completo do sistema, incluindo o eco das variáveis de comando e os valores de estado interno do PLC. A estrutura segue o padrão `success: true, data: ...`, e inclui cabeçalhos CORS para compatibilidade com clientes executados em *browsers*. O Listing 4.1 apresenta um exemplo de resposta.

Listing 4.1: Exemplo de resposta do `GET /api/status`

```
1 {  
2   "success": true,  
3   "data": {  
4     "state": 1,  
5     "currentPos": 47,  
6     "termo": false,  
7     "chuva": false,  
8     "mAuto": true,  
9     "mManual": false,  
10    "mEmergencia": false,  
11    "mSubir": false,  
12    "mDescer": false,  
13    "mSetPos": false,  
14    "mCalibrate": false,  
15    "targetPos": 50.0  
16  }  
17 }
```

O `POST /api/plc` recebe um pedido JSON com a estrutura `variable, value` e atualiza a *Shared Input* correspondente. O *sketch* valida o nome da variável, identifica o seu tipo e escreve o valor adequado, estando na resposta a confirmação da operação ou indicação de erro. Esta abordagem orientada a variáveis, em vez de a comandos de alto nível, reflete a natureza do mecanismo de *Shared Variables* e mantém o *sketch* flexível, dado que qualquer variável adicional pode ser exposta no futuro sem alterar a estrutura geral da interface.

4.4 Interface web

Foi desenvolvida uma interface web numa fase inicial, com o objetivo de explorar a comunicação com o autómato e validar a arquitetura proposta. A aplicação foi concebida para correr no computador do operador, que acede ao seu próprio servidor local através do navegador, devendo este estar ligado à mesma rede a que o autómato está ligado. No projeto, esta aplicação assumiu um papel exploratório, dado que serviu sobretudo para testar o servidor HTTP e validar a viabilidade da arquitetura REST proposta, bem como identificar possíveis limitações.

4.4.1 Enquadramento e tecnologia da interface web

Para construção da interface web foi utilizado o *framework* Next.js, assente na biblioteca *React*. É uma biblioteca *JavaScript* de código aberto, vocacionada para a construção de interfaces de utilizador a partir de componentes reutilizáveis. A abordagem baseada em componentes permite organizar a interface em blocos funcionais independentes, cada um responsável por uma parte da apresentação e da lógica associada, o que facilita tanto o desenvolvimento como a manutenção do código.

O Next.js é um *framework* que estende o React com funcionalidades de servidor, assentes num ambiente de execução Node.js. O Node.js é uma plataforma que permite executar código *JavaScript* fora do navegador, sendo então possível a construção de servidores web e serviços de *backend*. No contexto do Next.js, o Node.js é responsável por servir a aplicação, processar pedidos e executar lógica do lado do servidor. Entre as funcionalidades que o Next.js disponibiliza sobre esta base destacam-se as *API Routes*, que permitem implementar *endpoints* HTTP no próprio servidor da aplicação, sem necessidade de um serviço externo. Esta capacidade revelou-se determinante para a comunicação com o Opta, conforme descrito na subsecção seguinte. A Figura 4.6 apresenta a interface web desenvolvida.

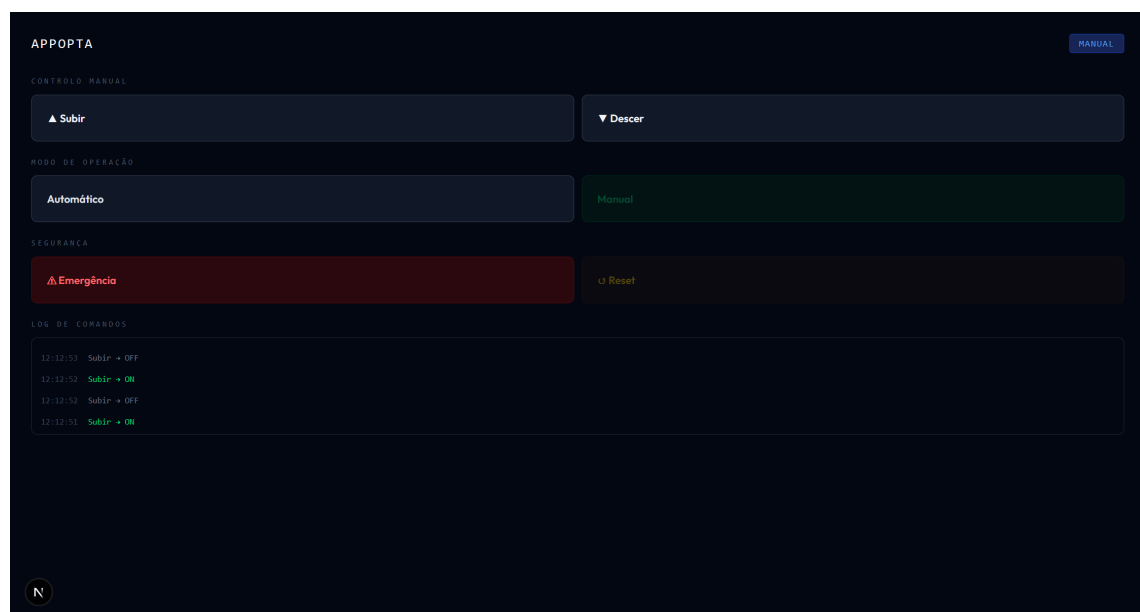


Figura 4.6: Interface web desenvolvida

4.4.2 Compatibilidade com o servidor desenvolvido

O desenvolvimento da interface *web* permitiu caracterizar o comportamento do servidor HTTP, quando acedido por clientes HTTP modernos, revelando duas incompatibilidades que decorrem diretamente da implementação minimalista descrita na secção anterior.

A primeira é a *same-origin policy* dos navegadores, tratando-se de um mecanismo de segurança que restringe a comunicação entre origens distintas. No cenário em causa, a página é carregada a partir do servidor local da aplicação Next.js e o código JavaScript executado no navegador tenta comunicar com o endereço de rede do Opta, o que resulta em duas origens diferentes do ponto de vista do navegador. Sem cabeçalhos *Cross-Origin Resource Sharing* (CORS) específicos nas respostas do autómato, o conteúdo das mesmas fica inacessível ao código cliente.

A segunda decorre da implementação `fetch` disponibilizada pelo Node.js, que recorre por defeito a funcionalidades do protocolo HTTP como o mecanismo de transferência em blocos (*Transfer-Encoding: chunked*) e as ligações persistentes (*Connection: keep-alive*). Estas funcionalidades pressupõem um servidor HTTP convencional, o que não corresponde à implementação do Opta, que espera pedidos com `Content-Length` explícito e encerra cada ligação após a resposta.

A solução adotada tem como objetivo adaptar o cliente às características conhecidas do servidor, sem modificar o *sketch*. Para isso foi implementado um servidor intermediário, *proxy* reverso, através das *API Routes* do Next.js, onde os pedidos do navegador passam a ser dirigidos a uma *API Route* da própria aplicação, que os reencaminha para o Opta e devolve a resposta. Como esta *API Route* corre no mesmo servidor que serve a página, a comunicação navegador-servidor ocorre sempre com a mesma origem, eliminando a restrição da *same-origin policy*. A *API Route* foi implementada com o módulo HTTP nativo do Node.js, em vez do `fetch`, o que permitiu fixar explicitamente os cabeçalhos `Connection: close` e `Content-Length`, garantindo compatibilidade com o formato de pedido esperado pelo servidor.

4.5 Padrões de comunicação

Antes de prosseguir para a descrição da aplicação móvel, importa descrever dois padrões de comunicação, uma vez que a aplicação móvel, descrita na Secção 4.6, reutiliza integralmente os mesmos mecanismos. Ambos os padrões foram desenvolvidos durante a implementação da interface web e decorrem diretamente da arquitetura do servidor HTTP descrita anteriormente. O primeiro padrão é a receção do estado do sistema por *polling* periódico ao *endpoint* `GET /api/status`. Em intervalos regulares de dois segundos, o cliente consulta o estado do Opta, processa a resposta em formato JSON e atualiza as variáveis internas que representam o estado do sistema. A deteção de transições entre o valor recebido e o valor anteriormente armazenado serve de base ao registo de eventos e à apresentação de informação atualizada ao utilizador. A escolha de *polling* em detrimento de mecanismos baseados em *push*, como *WebSockets* ou *Server-Sent Events*, é coerente com a implementação minimalista do servidor, que não suporta protocolos de comunicação bidirecional. O Algoritmo 4 sintetiza este ciclo.

Algorithm 4 Receção periódica do estado do Opta**Require:** Endereço do autómato, intervalo de amostragem T_{poll}

```

1: Estado: Modo, PosAtual, SensorChuva, SensorTermo

2: repeat
3:   aguardar  $T_{poll}$   $\triangleright T_{poll} = 2\text{ s}$ 
4:   enviar pedido HTTP GET a /api/status

5:   if comunicação falhou ou resposta inválida then
6:     sinalizar falha de comunicação
7:     continue
8:   end if

9:   extrair modorcv, posrcv, chuvarcv, termorcv da resposta JSON

10:  if algum dos valores recebidos diferir do estado interno then
11:    registar evento de transição correspondente
12:  end if

13:  Modo  $\leftarrow$  modorcv
14:  PosAtual  $\leftarrow$  posrcv
15:  SensorChuva  $\leftarrow$  chuvarcv
16:  SensorTermo  $\leftarrow$  termorcv
17: until aplicação terminada

```

O segundo padrão é o envio de comandos ao Opta através do *endpoint* POST */api/plc*, com o corpo do pedido em formato JSON da forma {*variable*, *value*}. Para os comandos associados aos botões de movimento manual, o valor é mantido a 1 enquanto o utilizador mantém a interação ativa, sendo reposto a 0 ao libertar, replicando o comportamento de um botão físico de pressão. Para os comandos numéricos, como a posição-alvo associada à variável *targetPos*, é transmitido diretamente o valor pretendido. Para os comandos por pulso, associados aos gatilhos do PLC, é necessária uma sequência de dois pedidos, existindo um primeiro que ativa a variável, e um segundo, separado por uma curta pausa, que a desativa, conforme apresentado no Algoritmo 5.

Algorithm 5 Envio de comando por pulso**Require:** *variavel*

```

1: Comum: envio de pedido HTTP POST a /api/plc com corpo JSON {variavel, valor}

2: enviar pedido com valor  $\leftarrow$  1
3: aguardar  $T_{pulso}$   $\triangleright T_{pulso} = 300\text{ ms}$ 
4: enviar pedido com valor  $\leftarrow$  0

```

O padrão de pulso aplica-se às variáveis *mAuto*, *mManual*, *mEmergencia*, *mSetPos* e *mCalibrate*. A pausa de 300 ms entre os dois pedidos foi determinada empiricamente, por ser suficiente para garantir que o ciclo de execução do PLC observe a transição ascendente da variável

e processe o gatilho, e é curta o suficiente para não introduzir latência perceptível na experiência do utilizador.

4.6 Aplicação móvel

Com a arquitetura de comunicação validada durante o desenvolvimento da interface web, foi desenvolvida uma aplicação móvel em *React Native*, com recurso ao *framework* Expo, destinada a ser a interface principal de operação do sistema. Por ambas as plataformas assentarem em React, os padrões de comunicação descritos na secção anterior foram reutilizados integralmente.

O React Native é uma plataforma de código aberto, que permite utilizar a mesma abordagem de componentes do *React* para construir aplicações móveis. Ao contrário de soluções baseadas em *WebView*, que embebem uma página web num invólucro nativo, esta plataforma compila os componentes para elementos nativos do sistema operativo, obtendo desempenho e comportamento tátil equivalente a uma aplicação desenvolvida nativamente. A partilha do modelo de programação com o *React* permitiu reaproveitar a lógica de comunicação com o Opta sem necessidade de a reimplementar numa linguagem ou paradigma diferente. O Expo é um *framework* que opera sobre o *React Native* e simplifica o processo de compilação e distribuição da aplicação, sendo possível compilar a aplicação através de um serviço remoto sem necessidade de configuração local do ambiente nativo.

Do ponto de vista da arquitetura de comunicação, a aplicação móvel apresenta uma diferença significativa face à web, dado que dispensa o servidor intermediário. As restrições de origem que motivaram aquela camada não se aplicam em ambiente nativo, uma vez que o conceito de origem do navegador não existe, visto que o código é executado diretamente pelo sistema operativo, não a partir de uma página servida por um anfitrião. Além disso, a implementação *fetch* disponibilizada pelo ambiente *React Native* não envia automaticamente os cabeçalhos de transferência em blocos que motivaram a adaptação técnica descrita anteriormente. Em consequência, a aplicação móvel comunica diretamente com o autómato, simplificando a arquitetura.

4.6.1 Funcionalidades Principais

No arranque da aplicação, e sempre que o Opta reporte um estado de não-inicialização, a aplicação apresenta um ecrã dedicado de inicialização, prevalecendo sobre qualquer outro ecrã. Este ecrã reflete o mecanismo de inicialização descrito anteriormente: o PLC, na ausência de feedback absoluto de posição, aguarda confirmação do operador sobre o estado físico real do equipamento antes de entrar em operação.

O ecrã apresenta uma mensagem clara a indicar que o sistema requer inicialização, acompanhada de dois botões visualmente distintos: *Fechado* e *Aberto*. A distinção visual entre os botões tem como propósito reduzir a probabilidade de uma seleção errada por parte do operador, dado que esta corromperia a estimativa inicial de posição. A mensagem inclui ainda uma nota informativa sobre a importância de observar o estado físico do *AirLight* antes da seleção.

Ao premir um dos botões, a aplicação emite o comando de movimento correspondente ao PLC, com o padrão de pulso habitual. O PLC interpreta o comando para efeitos de inicialização da estimativa, sem produzir movimento físico devido aos fins-de-curso, conforme detalhado na secção de controlo. Na resposta seguinte do ciclo de *polling*, a aplicação receberá a confirmação de que a inicialização ficou concluída e transita automaticamente para o painel principal, sem necessidade de intervenção adicional do utilizador. A Figura 4.7 apresenta este ecrã.



Figura 4.7: Ecrã de inicialização da aplicação móvel.

O ecrã principal constitui o painel de controlo central, apresentado ao utilizador no arranque da aplicação. O cabeçalho integra um indicador permanente do modo de operação em vigor, com código visual distinto para cada estado, e dois acessos: o botão de estatísticas e o botão de definições.

A área central do ecrã está organizada em quatro secções funcionais. As secções que não estão disponíveis no modo corrente são apresentadas com opacidade reduzida, comunicando a sua inatividade sem necessidade de mensagens de erro.

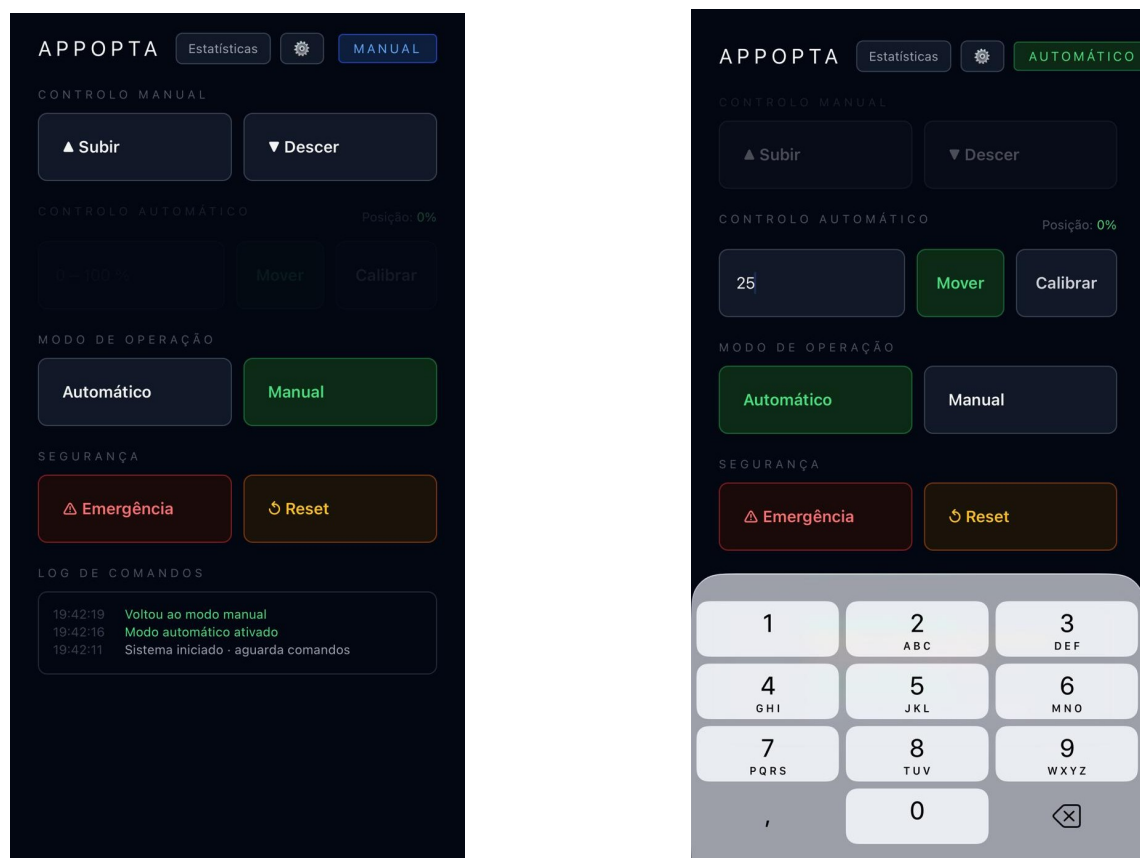
É possível alternar entre os modos disponíveis através dos botões, sendo que o correspondente modo em vigor é apresentado como destacado e inativo, prevenindo comutações redundantes. Em estado de emergência, ambos os botões ficam inativos, sendo apenas possível ativar o botão *Reset*, que repõe o sistema em modo manual após a resolução da ocorrência.

O controlo manual está disponível apenas no modo respetivo e contém dois botões de pressão contínua, Subir e Descer, sendo o movimento executado enquanto o utilizador mantém o dedo sobre o botão, cessando ao libertar. Quando o sensor de chuva está ativo, o botão Subir é adicionalmente desativado. Caso o sensor de temperatura seja ativado neste modo, a aplicação apresenta

um modal a sugerir a transição para o modo automático, conforme descrito mais à frente.

No modo automático é permitido ao utilizador introduzir uma posição-alvo entre 0 e 100, executada através do botão Mover. A posição atual do sistema é apresentada em tempo real. Esta página contém ainda o botão Calibrar, que aciona manualmente a rotina de calibração descrita anteriormente. Durante a sua execução, o botão fica visualmente assinalado como inativo e a aplicação bloqueia os restantes controlos até a rotina terminar, evitando que comandos do utilizador interfiram com a calibração em curso. A mesma sinalização é apresentada quando a rotina de calibração é desencadeada automaticamente pelo PLC, ao fim de quarenta movimentos. O estado é comunicado à aplicação através da *Shared Output* correspondente, sem que o utilizador tenha de pedir a operação explicitamente.

Neste modo, para além da funcionalidade da posição-alvo, o sistema continua responsável por responder aos valores do sensor de temperatura. Quando este se encontra ativo, é indicada visualmente essa condição, sendo a sinalização relevante porque a abertura do sistema pode estar a ser comandada autonomamente. As duas secções de controlo são apresentadas na Figura 4.8.



(a) Manual.

(b) Automático.

Figura 4.8: Secções principais de controlo na aplicação móvel.

4.6.2 Estatísticas e Modais de Aviso

O ecrã de estatísticas apresenta uma análise temporal do funcionamento do sistema, calculada a partir dos eventos persistidos no dispositivo. São apresentados três valores, formatados em horas e minutos: tempo total em estado aberto, tempo total em estado fechado e tempo total em movimento. É também possível observar o número total de comandos registados no dia, desdobrado em atuações originadas no utilizador e atuações ocorridas autonomamente pelo PLC em resposta aos sensores. A contagem de ativações de cada sensor e o número de avisos de temperatura aceites ou recusados pelo utilizador estão também presentes nesta secção, conforme ilustrado na Figura 4.9.



Figura 4.9: Secção de Estatísticas da aplicação móvel.

A aplicação interrompe a interação do utilizador, em situações específicas, através de modais que requerem resposta explícita antes de prosseguir. Estes modais partilham a estrutura visual: uma mensagem descritiva da situação e dois botões de ação, sendo o de aceitação acompanhado por um rótulo variável conforme o tipo de aviso. Apenas um modal pode estar ativo em simultâneo, sendo o próximo apresentado apenas após resposta ao modal corrente.

No modo manual, o aviso de sensor de temperatura surge quando o sensor é ativado estando o sistema neste modo de operação, o que leva a uma mensagem a questionar o utilizador se pretende mudar para o modo automático. A aceitação efetua essa transição imediatamente, enquanto a recusa mantém o sistema em modo manual e dispensa o aviso até que o sensor seja desativado.

No modo automático se o utilizador enviar o sistema para uma posição e o sensor de temperatura ficar ativo, é apresentada uma mensagem a questionar o utilizador se este prefere delegar

o controlo ao PLC, cancelando o comando de posicionamento, ou ignorar o aviso. Esta segunda opção leva ao envio do comando de bloqueio temporário do sensor de temperatura, através de `targetPos`, e em seguida envia a posição introduzida. A Figura 4.10 apresenta os dois modais.

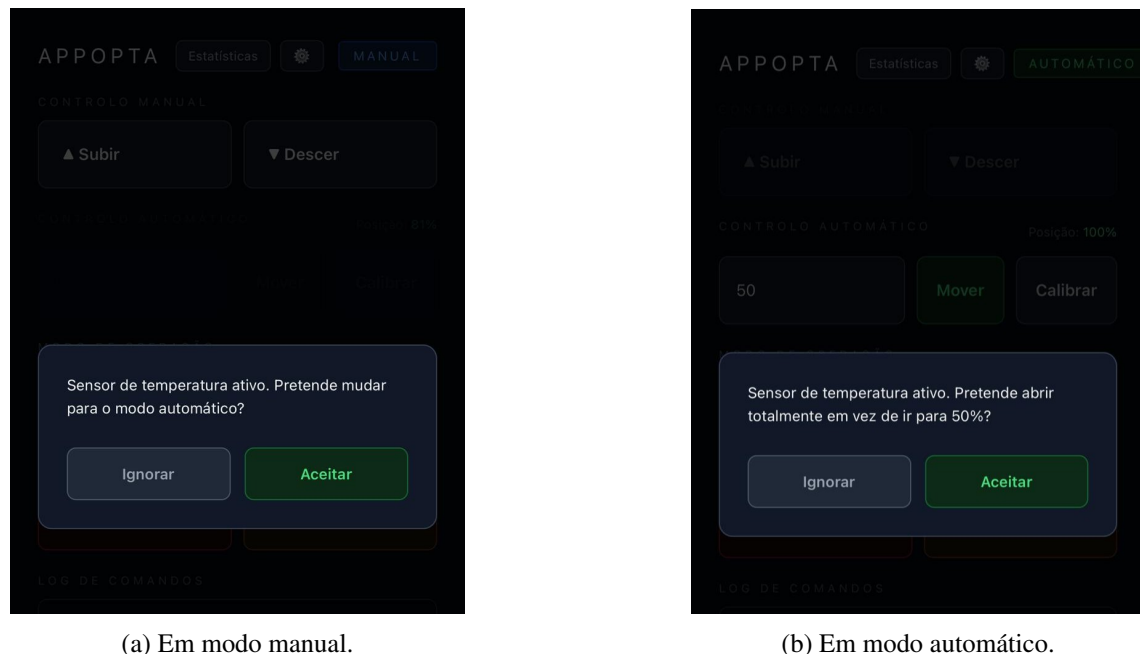


Figura 4.10: Modais relativos ao sensor de temperatura.

A aplicação regista, de forma estruturada, os eventos relevantes que ocorrem durante o seu funcionamento, armazenando-os localmente no dispositivo através do módulo *AsyncStorage*. Estes eventos têm duas origens distintas, ambas operando em paralelo durante a utilização da aplicação.

A primeira é a deteção de transições de estado durante o *polling* do Opta. Sempre que a leitura periódica do estado revela uma mudança significativa em relação à leitura anterior, um evento é registado. Esta deteção abrange mudanças de modo de operação, ativações dos sensores, e transições de posição (início de abertura, abertura completa, início de fecho, fecho completo). Adicionalmente, a aplicação infere o início e o fim do movimento físico do *AirLight*, sendo o início detetado quando a posição estimada começa a variar e o fim inferido quando a posição se mantém inalterada durante dois ciclos consecutivos de *polling*, correspondendo a aproximadamente quatro segundos sem alteração. A segunda origem remete para a ação direta do utilizador, sendo registado cada comando emitido e cada interação com os modais de aviso.

A partir deste registo, a aplicação calcula as métricas apresentadas no ecrã de estatísticas, sendo o cálculo dos tempos feito por emparelhamento de eventos complementares. Por exemplo, o tempo em estado aberto resulta da soma dos intervalos entre cada evento de início de abertura e o evento seguinte de fecho completo. Este método permite reconstruir uma caracterização contínua do comportamento do sistema a partir de eventos discretos.

4.6.3 Compilação e distribuição da aplicação

A execução da aplicação durante o desenvolvimento está dependente do servidor local do *Expo*, que serve o código *JavaScript* em tempo real ao dispositivo. Este modo de funcionamento implica que o ambiente de desenvolvimento esteja ativo no computador, tornando a aplicação dependente dessa infraestrutura. Para utilização operacional autónoma do dispositivo móvel, foi gerada uma versão compilada e empacotada num ficheiro APK (*Android Package*), instalável diretamente em qualquer dispositivo Android sem necessidade de qualquer servidor de desenvolvimento ligado.

A geração deste ficheiro foi realizada através do serviço *EAS Build*, a solução oficial do ecossistema *Expo* para a compilação de aplicações *React Native*. Este serviço executa a compilação em servidores remotos geridos pela *Expo*, dispensando a instalação local do conjunto de ferramentas de desenvolvimento *Android*.

No processo de configuração foi necessário acomodar uma particularidade técnica relacionada com a arquitetura de comunicação do sistema. As versões modernas de *Android* bloqueiam por omissão o tráfego de rede em HTTP não cifrado, permitindo apenas comunicação via HTTPS. Uma vez que o servidor HTTP incorporado no *sketch* é uma implementação minimalista que não suporta HTTPS, e que a comunicação com o autómato ocorre por HTTP simples no endereço de rede local, foi necessário declarar explicitamente a permissão de tráfego não cifrado na configuração da aplicação Android, através da propriedade `usesCleartextTraffic`. Sem esta declaração, a aplicação instalada como APK autónomo seria incapaz de comunicar com o Opta, com os pedidos a ser silenciosamente bloqueados pelo sistema operativo.

4.7 Integração com sistemas através de Modbus

A interface REST descrita anteriormente constitui o canal de comunicação principal do sistema com aplicações web, porém, numa perspetiva de integração com outros sistemas industriais, onde os protocolos normalizados são dominantes, a interface REST não é a opção mais adequada, conforme discutido no Capítulo 2. Desta forma, um dos objetivos na fase de desenvolvimento foi, também, perceber de que forma poderíamos explorar a capacidade de comunicação do autómato baseada no protocolo *Modbus* TCP através da sua porta RJ-45.

Para validar a viabilidade desta vertente foi conduzida, durante uma fase inicial do projeto, uma experiência de comunicação entre o Opta e um cliente externo, com o autómato a desempenhar o papel de *slave*. Esta experiência teve como objetivo permitir confirmar a capacidade do equipamento para se integrar em redes industriais via Modbus TCP, e estabelecer o ponto de partida para integrações futuras com sistemas como BMS industriais, SCADA ou equipamentos HVAC.

A experiência foi realizada ainda com a variante do controlador sem conectividade Wi-Fi (AFX00001), que dispõe apenas de interface Ethernet, e com um computador, responsável por

desempenhar o papel de *master*. A configuração da rede no Opta foi feita num bloco de parâmetros do Arduino PLC IDE, à semelhança do bloco Wi-Fi descrito anteriormente, sendo atribuído um endereço IP estático ao equipamento, conforme demonstra a Figura 4.11. A funcionalidade de servidor Modbus TCP é suportada nativamente pelo *runtime* do Arduino PLC IDE, sem necessidade de *sketch* adicional para gerir o protocolo, sendo o servidor iniciado automaticamente quando existem variáveis declaradas como acessíveis via *Modbus*.

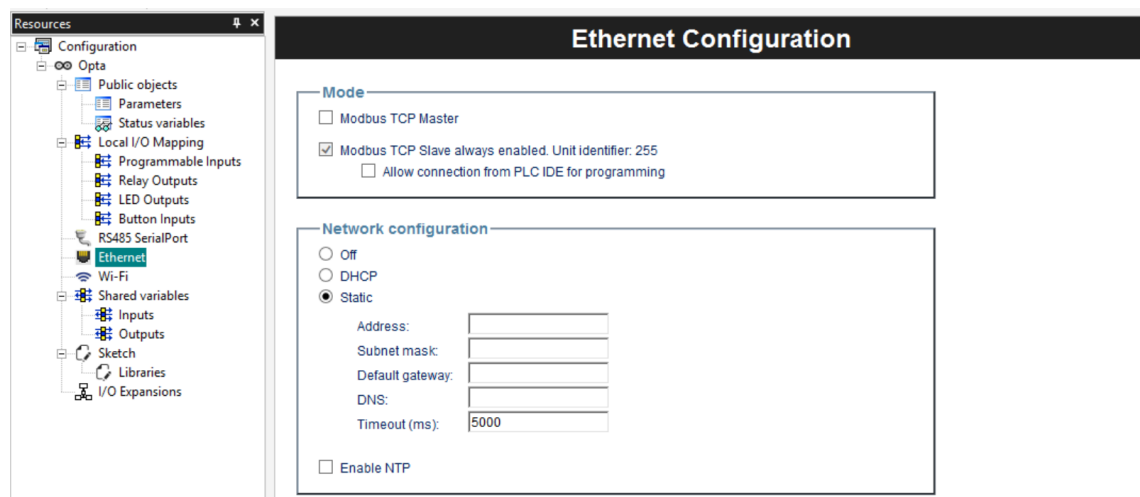


Figura 4.11: Configuração de parâmetros Ethernet no Arduino PLC IDE.

Do lado do computador, foi utilizado o *qModMaster*, um simulador que permite construir pedidos de leitura e escrita configurando o endereço IP do *slave*, o *function code* correspondente à operação pretendida e o endereço da variável a aceder. Recorrendo a este simulador, foi possível ler e escrever variáveis no PLC em tempo real, validando a comunicação bidirecional entre os dois sistemas. A Figura 4.12 apresenta a interface do simulador durante um destes testes.

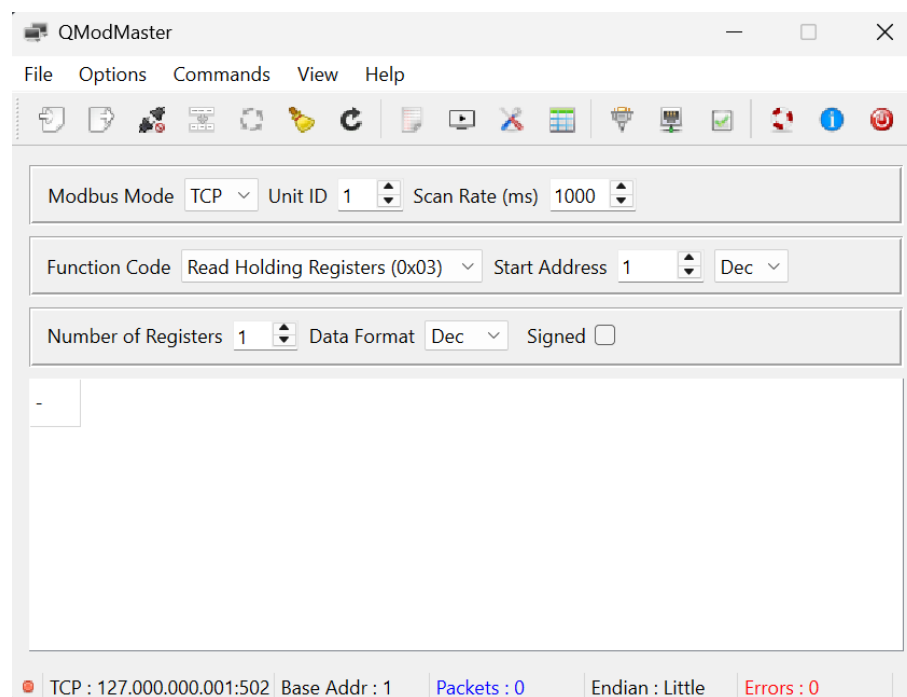


Figura 4.12: Simulador qModMaster.

Existe uma diferença relevante face à arquitetura final do sistema no que respeita à declaração das variáveis expostas ao exterior. Enquanto na implementação por REST a troca de informação entre o *sketch* e o PLC é assegurada pelas *Shared Variables*, no caso do Modbus TCP as variáveis acessíveis ao exterior são declaradas como *Parameters*. Este tipo de variável é especificamente projetado para exposição via *Modbus*, sendo o respetivo endereço atribuído automaticamente pelo Arduino PLC IDE e consultável através de uma tabela disponibilizada no ambiente de desenvolvimento. Cada variável fica, deste modo, associada a um endereço único que pode ser referenciado pelo *master* para realizar operações de leitura ou escrita, conforme visível na Figura 4.13.

#	Address (dec)	Address (hex)	Name	Parameter type	PLC type	Size	Default value	Min
1	16384	0x4000	mSubir	Boolean	BOOL		False	
2	16385	0x4001	mDescer	Boolean	BOOL		False	
3	16386	0x4002	mAuto	Boolean	BOOL		False	
4	16387	0x4003	mManual	Boolean	BOOL		False	
5	16388	0x4004	mEmergencia	Boolean	BOOL		False	
6	16389	0x4005	targetPos	Real	REAL		0	
7	16391	0x4007	mSetPos	Boolean	BOOL		False	
8	16392	0x4008	mCalibrar	Boolean	INT		False	
9	16393	0x4009	mState	Signed 16-bit	INT		0	
10	16394	0x400A	mCurrentPos	Real	REAL		0	
11	16396	0x400C	mchuva	Boolean	BOOL		False	
12	16397	0x400D	mtermo	Boolean	BOOL			

Figura 4.13: Declaração de variáveis como *Parameters*.

Embora limitada em âmbito, esta experiência valida três aspetos importantes para a perspetiva de integração industrial. Em primeiro lugar, demonstra que o Opta pode operar como *slave*

Modbus sem necessidade de *hardware* adicional ou de *middleware* de tradução, recorrendo apenas às funcionalidades nativas do Arduino PLC IDE. Em segundo lugar, mostra que o mecanismo de *Parameters* é simples e direto, permitindo expor ao exterior qualquer variável do programa PLC com um endereço Modbus bem definido. Em terceiro lugar, valida que a infraestrutura de rede envolvida corresponde ao padrão típico de redes industriais Ethernet, com endereçamento IP normalizado e porta TCP/IP padronizada (502).

A partir desta base, a integração futura com sistemas industriais torna-se uma extensão natural da arquitetura desenvolvida. Nestas integrações, o sistema externo assume o papel de *master* Modbus e o Opta expõe um conjunto de registos correspondentes aos parâmetros e comandos relevantes do *AirLight*, permitindo a sua leitura e escrita de forma normalizada. A coexistência entre a interface REST e o servidor Modbus no mesmo equipamento é possível, dado que ambos operam em portas distintas e não interferem entre si, permitindo que o sistema sirva simultaneamente clientes de aplicações de alto nível e clientes industriais.

4.8 Implementação experimental de BMS com Home Assistant

A presente secção descreve a implementação experimental de uma camada de supervisão sobre o sistema desenvolvido, com o objetivo de validar a aplicação das estratégias discutidas no Capítulo 2. A descrição parte da escolha e justificação da plataforma utilizada, segue para a arquitetura de comunicação que a liga ao controlador local e para a caracterização das fontes de informação que alimentam as automações, culminando na apresentação das estratégias de supervisão efetivamente implementadas.

4.8.1 Enquadramento e escolha da plataforma

Para validar a integração do sistema desenvolvido numa camada de supervisão, foi implementado um sistema de gestão técnica sobre a plataforma *Home Assistant*. Esta plataforma, de código aberto, foi originalmente concebida para automação residencial, mas a sua arquitetura modular e o suporte nativo para múltiplos protocolos de comunicação, como HTTP/REST, MQTT e até Modbus, tornam-na adequada para reproduzir o papel de um sistema de gestão de edifícios num contexto experimental.

Importa clarificar que o *Home Assistant* não é um sistema de gestão de edifícios certificado para uso industrial, é utilizado aqui como plataforma de supervisão capaz de demonstrar a lógica de gestão pretendida. O objetivo do trabalho descrito nesta secção não é a validação de um produto BMS comercial, mas a demonstração de que o sistema pode ser coordenado por uma camada de supervisão superior que executa estratégias de gestão ambiental, estratégias essas fundamentadas na revisão de literatura apresentada no Capítulo 2.

4.8.2 Arquitetura de integração

A comunicação entre o *Home Assistant* e o Opta recorre à mesma interface REST descrita anteriormente, confirmando que esta interface serve simultaneamente as aplicações de utilizador final e a camada de supervisão.

No sentido da leitura, o *Home Assistant* consulta periodicamente o estado do Opta, através de um pedido ao *endpoint* `/api/status` a cada cinco segundos. A resposta JSON é processada para extrair os valores relevantes como a posição atual, o modo de operação e o estado dos sensores de chuva e de temperatura, que ficam disponíveis como entidades da plataforma, passíveis de serem usadas em automações e visualizações.

No sentido da escrita, o *Home Assistant* emite comandos para o Opta através de pedidos ao *endpoint* `/api/plc`. Estes comandos reproduzem a mesma lógica de sequência de pulsos utilizada pelas aplicações, sendo estas sequências encapsuladas em rotinas reutilizáveis na plataforma, correspondentes às operações de alto nível necessárias às estratégias de supervisão: colocar o sistema em modo automático, abrir totalmente, fechar totalmente, ou posicionar a meia abertura.

Desta forma, o *Home Assistant* assume o papel de camada de supervisão que coordena o controlador local, sem substituir a sua lógica de controlo determinístico. O autómato mantém a responsabilidade pela execução do movimento e pela gestão dos modos de operação, limitando-se a camada de supervisão a emitir comandos de alto nível em função das estratégias de gestão definidas, deixando ao controlador local a sua concretização.

4.8.3 Modelação das entradas

As estratégias de supervisão dependem de informação sobre o ambiente e a ocupação do espaço. No sistema implementado, esta informação provém de três categorias de fontes distintas, cuja natureza importa distinguir com clareza.

A primeira categoria é o estado do próprio sistema controlado, obtido em tempo real através da interface REST. Inclui a posição atual e o estado dos seus sensores, sendo esta informação real e reflete o estado efetivo do equipamento.

A segunda categoria associa-se a um conjunto de sensores ambientais internos que representam as seguintes variáveis: concentração de CO₂, luminosidade, temperatura interior e exterior, ocupação do espaço, e estado dos sistemas de climatização. Estas entradas foram modeladas através de entidades de valor manual do *Home Assistant*, cujos valores são definidos diretamente para efeitos de teste das estratégias de supervisão. Esta opção é intencional e decorre do âmbito do trabalho, sendo o foco a validação da lógica de supervisão e a sua articulação com o sistema controlado, não na aquisição de dados ambientais reais. Num cenário de produção, estas entradas seriam substituídas por sensores físicos ou por integrações com sistemas de aquisição existentes no edifício. As automações implementadas dependem apenas dos valores das entidades, sendo independentes da origem desses valores, pelo que a substituição não exigiria alterações à lógica de supervisão.

A terceira categoria é a informação meteorológica, obtida de uma fonte real através da integração com o serviço Met.no, disponível nativamente no Home Assistant. A partir da previsão horária desta fonte, são derivados dois indicadores usados nas estratégias preventivas: a previsão de precipitação nas duas horas seguintes e a temperatura máxima prevista para as oito horas seguintes. O recurso a uma fonte meteorológica real demonstra que a substituição de entradas simuladas por fontes reais é direta e foi efetivamente concretizada para esta categoria de informação. A Figura 4.14 apresenta a interface do Home Assistant com as entidades configuradas.

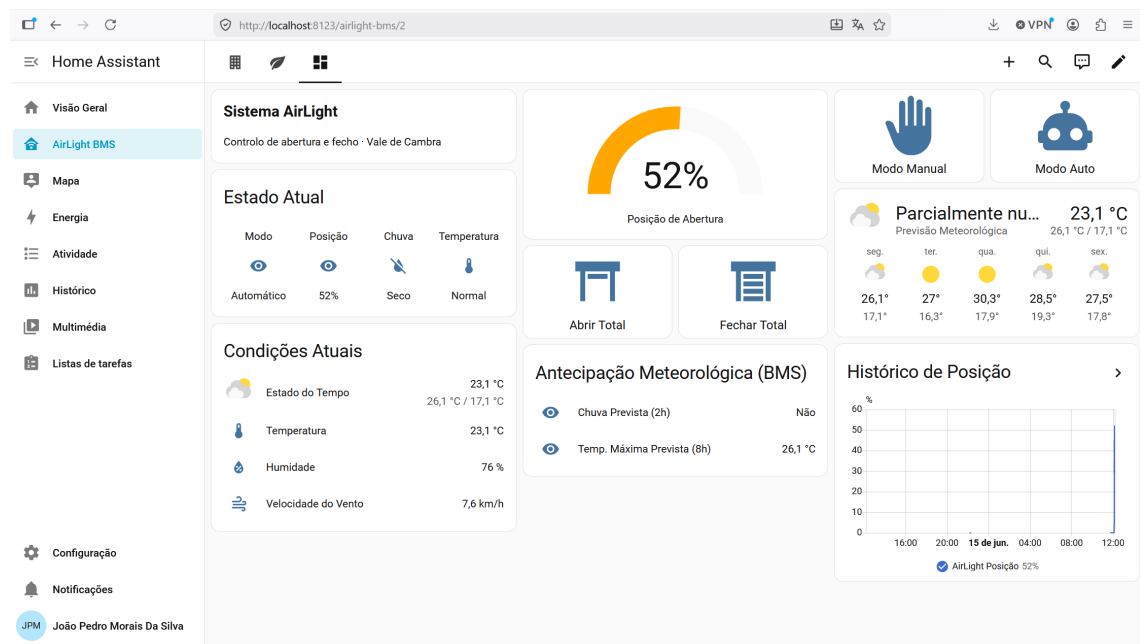


Figura 4.14: Interface Home Assistant.

4.8.4 Estratégias de supervisão implementadas

As estratégias de supervisão foram implementadas como automações na plataforma em estudo, cada uma traduzindo uma regra de gestão ambiental num conjunto de condições e ações sobre o sistema. Apresentam-se, de seguida, as estratégias implementadas, organizadas pela natureza do objetivo que servem, e fundamentadas nos valores e referências discutidos no Capítulo 2.

A primeira estratégia refere-se ao arrefecimento noturno (*free cooling*), e que visa aproveitar o ar exterior mais fresco durante a noite para arrefecer o espaço sem recurso a climatização ativa. É ativada de noite ou de madrugada, quando a temperatura interior excede os 24 °C, e o diferencial entre a temperatura interior e a exterior é de pelo menos 3 °C. Os dois primeiros limiares correspondem ao limite superior da faixa de conforto da Categoria II da norma EN 16798-1, já o diferencial de 3 °C corresponde ao valor estabelecido segundo a revisão de literatura, como condição mínima para a eficácia da ventilação noturna. Quando as condições se verificam, a camada de supervisão coloca o sistema em modo automático e comanda a abertura total do *AirLight*.

A ventilação por previsão de calor, é complementar à estratégia anterior, dado que esta antecipa períodos de calor recorrendo à previsão meteorológica. É ativada no início da manhã quando a

temperatura máxima prevista para as oito horas seguintes excede os 30 °C, desde que não esteja a chover e o ar condicionado não esteja em funcionamento. Nessas condições, o sistema é colocado em modo automático e o *AirLight* é aberto, promovendo a renovação e o arrefecimento do espaço antes da chegada do período mais quente. A utilização de previsão meteorológica real, proveniente da integração Met.no, permite a esta estratégia atuar de forma preventiva em vez de reativa.

A renovação de ar por concentração de CO₂, assegura a qualidade do ar interior, monitorizando a concentração de dióxido de carbono como indicador indireto da ocupação e da necessidade de renovação. É ativada quando a concentração de CO₂ excede os 1200 ppm, na presença de ocupação detetada e desde que o sensor de chuva não esteja ativo. O limiar de 1200 ppm corresponde à Categoria II da norma EN 16798-1, conforme apresentado no Capítulo 2. Verificadas as condições, o sistema é colocado em modo automático e é aberto até metade, promovendo a renovação do ar sem abertura total desnecessária.

A iluminação inteligente é responsável por gerir a iluminação artificial em função da luz natural disponível, evitando consumo desnecessário. Quando a luminosidade interior desce abaixo de 400 lux na presença de ocupação, a iluminação artificial é ativada. Por outro lado, quando a luminosidade ultrapassa 700 lux, a iluminação artificial é desligada. Estes limiares enquadram-se nos valores de referência para iluminância em espaços de trabalho, e a estratégia ilustra o princípio de aproveitamento da luz natural, *daylight harvesting*, discutido no Capítulo 2. No caso de se usarem LEDs dimáveis, a estratégia poderia ser adaptada para que a potência destes fosse ajustada em função da luz natural, em vez de iluminação artificial ser só desligada ou ligada. É de destacar que, ao contrário das restantes estratégias, esta atua sobre a iluminação e não diretamente sobre o *AirLight*. A captação de luz natural é assegurada pela cúpula, fixa em relação à estrutura superior do equipamento, pelo que o movimento de abertura, sendo elevatório, não altera a área nem o ângulo de exposição ao céu. A posição do *AirLight* influencia apenas ligeiramente a luz natural disponível no espaço, através de alguma luz indireta admitida pelo vão lateral quando o sistema se encontra aberto.

A coordenação entre climatização ativa e ventilação natural previne o desperdício energético que ocorre quando o ar condicionado e o *AirLight* operam em simultâneo, conforme ilustrado na Figura 4.15. Desta forma, quando o ar condicionado é ligado e o *AirLight* se encontra aberto, a camada de supervisão coloca o sistema em modo automático e comanda o fecho total do sistema, evitando que o espaço climatizado perca o ar arrefecido para o exterior. Esta coordenação reflete uma boa prática de gestão técnica de edifícios, em que sistemas ativos e passivos são geridos de forma articulada para evitar perdas de energia.

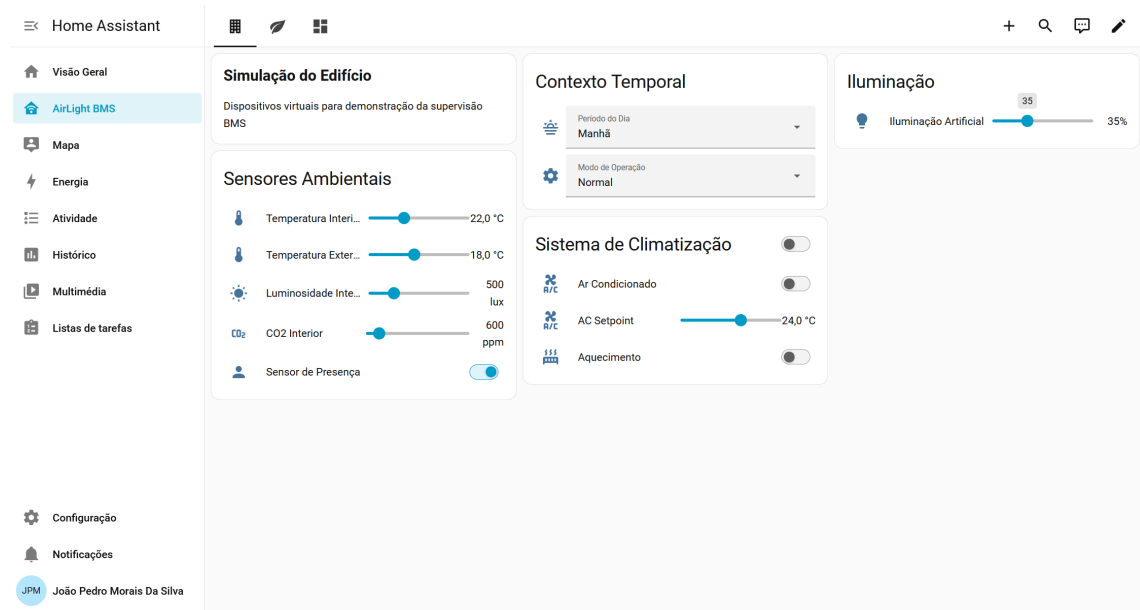


Figura 4.15: Estratégias de supervisão e estado global do edifício.

Em complemento a esta estratégia preventiva, foi implementada uma detecção passiva da mesma ineficiência. Um indicador de supervisão monitoriza continuamente a coexistência de ar condicionado em funcionamento com o *AirLight* aberto, sinalizando essa condição como ineficiência detetada. Esta dupla abordagem, atuação preventiva e monitorização passiva, assegura que a ineficiência é não só evitada ativamente, mas também detetada caso ocorra por alguma via não prevista, como uma intervenção manual posterior do utilizador. De forma análoga, um segundo indicador deteta a presença de iluminação artificial ligada na ausência de ocupação, sinalizando outra forma de desperdício energético. A Figura 4.16 apresenta estes indicadores.

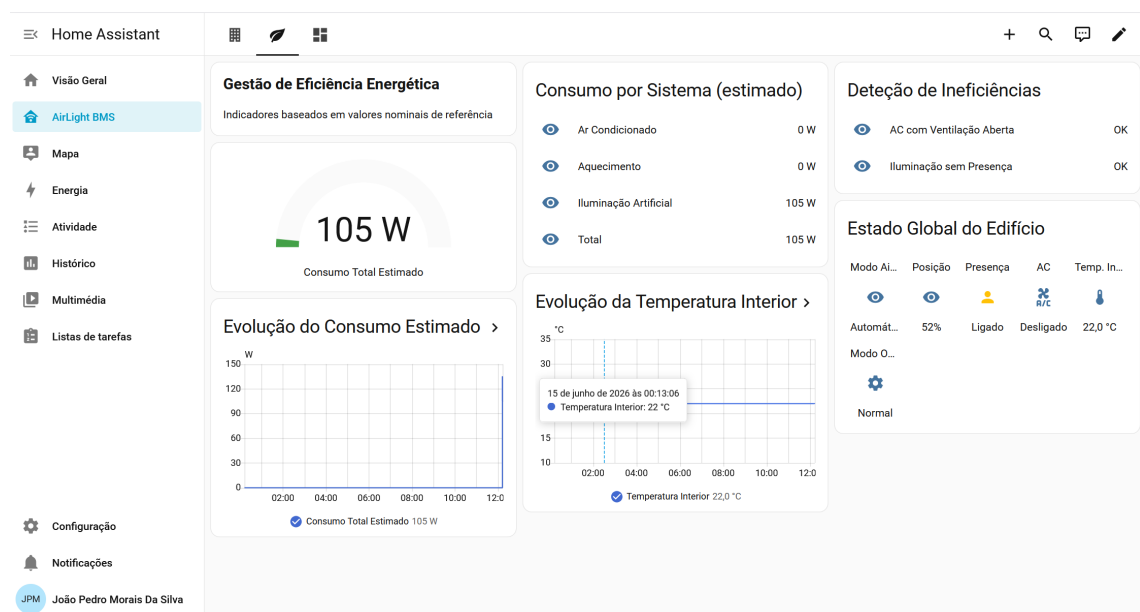


Figura 4.16: Indicadores de consumo e detecção de ineficiências.

Foi, também, definida uma estratégia que coloca o sistema num estado seguro e eficiente quando o espaço deixa de estar ocupado. Quando a ocupação deixa de ser detetada, a camada de supervisão fecha totalmente o *AirLight*, desliga a iluminação artificial e desliga o ar condicionado, garantindo que nenhum sistema permanece ativo desnecessariamente num espaço desocupado.

Da mesma forma que na ventilação preventiva, explicada anteriormente, foi também usada uma estratégia que assume o papel de proteger o espaço interior contra a entrada de água, recorrendo à previsão meteorológica. Quando é prevista precipitação nas duas horas seguintes e o *AirLight* se encontra aberto, o sistema é colocado em modo automático e fechado totalmente. Tal como na ventilação preventiva por calor, esta estratégia beneficia da previsão meteorológica real proveniente da integração Met.no, atuando de forma antecipada à ocorrência da precipitação.

Estas estratégias coexistem e podem, em determinadas circunstâncias, produzir comandos concorrentes. A sua articulação assenta na priorização implícita das condições, dado que, por exemplo, a proteção contra chuva sobrepõe-se à renovação de ar e na natureza determinística do controlador local, que processa os comandos recebidos de acordo com a sua própria máquina de estados. A gestão de conflitos entre estratégias de forma mais sofisticada, com uma hierarquia de prioridades explícita, constitui uma das vias de evolução identificadas para trabalho futuro. Importa também notar que a funcionalidade de resposta automática ao sensor de temperatura do próprio *AirLight*, opera de forma independente das estratégias de supervisão aqui apresentadas e não foi integrada com a camada de BMS. Esta funcionalidade reage apenas ao estado da entrada binária associada ao sensor, sem avaliar se a abertura corresponde efetivamente a uma estratégia de arrefecimento eficaz face às condições exteriores presentes. Distintamente da ventilação preventiva por calor previsto, que se baseia em temperaturas previstas, esta avaliação exigiria a medição em tempo real de ambas as temperaturas, interior e exterior, requisito de instrumentação que não foi contemplado no âmbito do presente trabalho, sendo retomado na Secção 6.2.

4.9 Estudo de Integração com Central de Desenfumagem

Conforme descrito no Capítulo 3 a integração com a central de desenfumagem foi estudada como vertente exploratória, não fazendo parte da configuração base do sistema. As alterações ao *software* descritas nesta secção foram implementadas e testadas nesse contexto, demonstrando a viabilidade da coexistência, mas não estão ativas no sistema de controlo que constitui o resultado principal desta dissertação. Esta integração exigiu uma articulação entre o atuador, a central e o Opta, sendo este último responsável por comandar a entrada de conforto da central, com a central a assumir o comando direto dos atuadores em situação de alarme. Esta arquitetura, exigiu alterações específicas tanto na lógica do PLC como no servidor HTTP implementado no sketch. Esta secção descreve essas alterações concretas. A integração introduziu duas variáveis novas no programa ModeMachine.

A primeira, *alarmeOk*, corresponde à entrada digital I8, ligada ao terminal de sinalização de alarme da central. Em condições normais, este terminal entrega a tensão necessária ao Opta,

ficando a variável com sinal lógico positivo, porém, em situação de alarme, o sinal é cortado, negando o valor lógico da variável.

A segunda variável, *incendio*, é derivada diretamente de *alarmeOk* no início de cada ciclo de execução, sendo o seu valor o complemento lógico da primeira, que reflete o estado físico do sinal, enquanto a segunda reflete a interpretação semântica desse sinal pelo programa. A utilização de *incendio* no resto da lógica torna a leitura do código mais próxima da intenção, sem perder o significado *fail-safe*.

Foi adicionado um novo estado ao *ModeMachine*, de valor numérico 3, simbolizando o estado de alarme e que se sobrepõe aos restantes modos quando ativo. A transição para este estado é forçada incondicionalmente, qualquer que seja o estado anterior. A detecção de alarme desencadeia a passagem imediata para o estado 3, garantindo que o sistema responde ao alarme sem latência. O Algoritmo 6 apresenta esta lógica.

Algorithm 6 Tratamento do estado de alarme no PLC

Require: *alarmeOk*

```

1: Estado: State, incendio, CmdSubir, CmdDescer, SetPosActive

2: incendio  $\leftarrow \neg \text{alarmeOk}$ 

3: if incendio then
4:   State  $\leftarrow$  ALARME
5: end if

6: if State = ALARME then
7:   CmdSubir  $\leftarrow$  false
8:   CmdDescer  $\leftarrow$  false
9:   SetPosActive  $\leftarrow$  false

10:   alternar estado dos LEDs com cadência rápida

11:   if alarmeOk then
12:     State  $\leftarrow$  MANUAL
13:   end if
14: end if
  
```

No estado de alarme, os comandos de movimento do autômato são bloqueados, quer os manuais (botões e aplicação móvel), quer os de posicionamento automático. Esta inibição é coerente com a arquitetura escolhida, visto que durante o alarme, é a central de desenfumagem que comanda diretamente os atuadores, não cabendo ao Opta emitir comandos concorrentes. A inibição é total e independente do modo anterior, porque, em alarme, o autômato recua para um papel exclusivamente observacional.

A sinalização visual é assegurada pelo piscar dos LEDs do dispositivo, comandados pelo temporizador *BlinkTimerA* configurado com período de 200 ms. Esta cadência é deliberadamente mais rápida que a do estado de Emergência (500 ms), criando uma distinção visual imediata entre

os dois modos de exceção do sistema. A saída do estado de Alarme ocorre quando `alarmeOk` regressa ao valor de 1, ou seja, quando a central liberta o sinal. Adotou-se uma estratégia em que o sistema transita para o modo manual (Estado 0) após a saída do alarme, em vez de regressar ao modo em que se encontrava previamente. Esta escolha privilegia a intervenção consciente do operador na sequência de uma ocorrência de segurança, em detrimento da retoma automática de operação que poderia ser desadequada após o evento. Para além do novo estado, foi necessário ajustar pontualmente os estados existentes, nomeadamente, a lógica de resposta ao sensor de chuva, presente no estado manual e automático, para que a transição forçada para o estado de alarme tenha precedência sobre essa resposta, garantindo que nenhum movimento desencadeado por chuva possa coincidir com uma situação de alarme em curso.

Relativamente ao servidor HTTP implementado no *sketch*, foi necessário expor a variável `incendio` na resposta JSON do GET `/api/status`, à semelhança das restantes variáveis de estado. Esta exposição permite que os clientes externos conheçam em tempo real a situação de alarme do sistema e adaptem o seu comportamento, por exemplo, apresentando uma sinalização visual destacada ao utilizador. Foi, necessário alterar funcionalidades no que diz respeito à recusa de comandos durante o alarme. Quando estamos em caso de incêndio, o *endpoint* POST `/api/plc` rejeita todos os pedidos de escrita às variáveis de comando, devolvendo um código de estado HTTP 400 com uma indicação textual da causa. Esta política é coerente com a inibição equivalente que ocorre no PLC, sendo a central responsável pelo controlo durante o alarme, porque não é desejável que comandos externos sejam aceites, ainda que apenas para serem ignorados internamente. A recusa explícita ao nível da API torna o comportamento mais previsível para os clientes e evita a ambiguidade de comandos aparentemente aceites, mas sem efeito.

4.10 Síntese do Desenvolvimento

O trabalho descrito neste capítulo resultou num sistema que integra a lógica de controlo do equipamento, uma arquitetura de comunicação baseada em REST sobre HTTP e uma plataforma de supervisão experimental. A lógica implementada no autómato organiza o controlo em três programas com responsabilidades distintas, suportando múltiplos modos de operação, posicionamento por percentagem e calibração automática. A interface de comunicação desenvolvida serve simultaneamente as aplicações de utilizador e a plataforma de supervisão, tendo a viabilidade de integração com sistemas industriais sido demonstrada através de uma prova de conceito baseada em Modbus TCP. A plataforma de supervisão, implementada sobre o Home Assistant, demonstrou a aplicação prática das estratégias identificadas na revisão da literatura, recorrendo a previsões meteorológicas reais.

O capítulo seguinte apresenta a validação experimental do sistema, incidindo sobre o desempenho da arquitetura de comunicação e a estabilidade dos tempos de curso do atuador.

Capítulo 5

Resultados Experimentais

O presente capítulo apresenta os resultados obtidos na validação experimental do sistema de controlo desenvolvido. A demonstração integral do funcionamento do *AirLight* e da sua lógica de controlo, bem como a articulação com estratégias de supervisão, é de natureza essencialmente dinâmica e visual, sendo por isso mais adequadamente percebida através de vídeo, a apresentar na defesa pública da dissertação. Optou-se, então, por documentar neste capítulo os resultados que são possíveis de justificar e analisar de forma rigorosa em formato de relatório: o desempenho da arquitetura de comunicação implementada, e a caracterização dos tempos de curso do atuador ao longo de um conjunto extenso de ciclos de operação.

5.1 Caracterização do desempenho de comunicação e resposta

Para caracterizar o desempenho do sistema em condições reais de utilização, foram realizados dois ensaios experimentais em duas redes Wi-Fi distintas (doméstica e empresarial). O primeiro ensaio mede a latência de comunicação pura, utilizando um comando lógico sem movimento, já o segundo mede o tempo total de resposta, incluindo o movimento mecânico. A combinação dos dois ensaios permite separar conceptualmente o tempo de comunicação do tempo de atuação física, dando uma leitura mais rigorosa do comportamento do sistema. As visualizações apresentadas nesta secção foram geradas a partir dos dados experimentais recolhidos, com o código de processamento e representação gráfica desenvolvido com o apoio de ferramentas de inteligência artificial.

5.1.1 Latência de Comunicação

O primeiro ensaio mede o tempo de ida-e-volta (*round-trip time*) de um pedido HTTP POST ao *endpoint* do OPTA, desde o envio até à receção da resposta de confirmação. Foi utilizado o comando de alteração de modo de operação, alternando entre o modo manual e automático, por ser um comando puramente lógico que não desencadeia movimento físico. Esta escolha isola a latência da camada de comunicação e do processamento interno do OPTA, sem contaminação por atuação mecânica. Foram recolhidas 150 amostras por rede, com um *script* externo a executar

pedidos com cadência fixa de 300 ms e *timeout* de 5 segundos. A Tabela 5.1 resume os resultados obtidos.

Tabela 5.1: Latência de comunicação entre cliente e OPTA, por rede.

Parâmetro	Rede doméstica	Rede da empresa
Mediana (ms)	314	113
Média (ms)	360	174
Desvio padrão (ms)	247	293
Percentil 95 (ms)	524	242
Mínimo (ms)	83	61
Máximo (ms)	2897	2295
N	150	150

A rede da empresa apresenta uma latência mediana cerca de três vezes inferior à da rede doméstica. A distribuição dos valores na empresa concentra-se entre 60 e 200 ms, enquanto a distribuição na rede doméstica apresenta um pico dominante entre 300 e 320 ms. O desvio padrão da rede da empresa surge aparentemente mais elevado, mas esta leitura é enganadora: deve-se à presença de quatro amostras esporádicas superiores a 1000 ms num universo de 150 medições. Excluindo estes *outliers*, o desvio padrão típico é de 116 ms na rede doméstica e de 46 ms na rede da empresa, confirmando que esta última é simultaneamente mais rápida e mais consistente. Os picos esporádicos observados são característicos de comunicação Wi-Fi, retransmissões, congestão momentânea e interferências, tendo sido mantidos na análise por serem fenómenos reais do meio de comunicação. A distribuição das latências é visível no histograma da Figura 5.1.

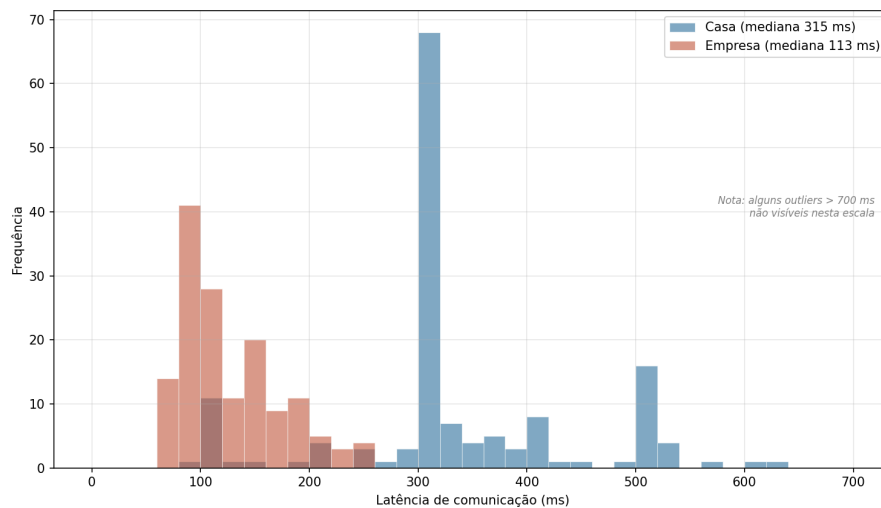


Figura 5.1: Histograma da latência de comunicação do comando de alteração de modo, por rede.

A Figura 5.2 compara as duas redes em formato de *boxplot*, e a Figura 5.3 apresenta a evolução da latência ao longo dos pedidos.

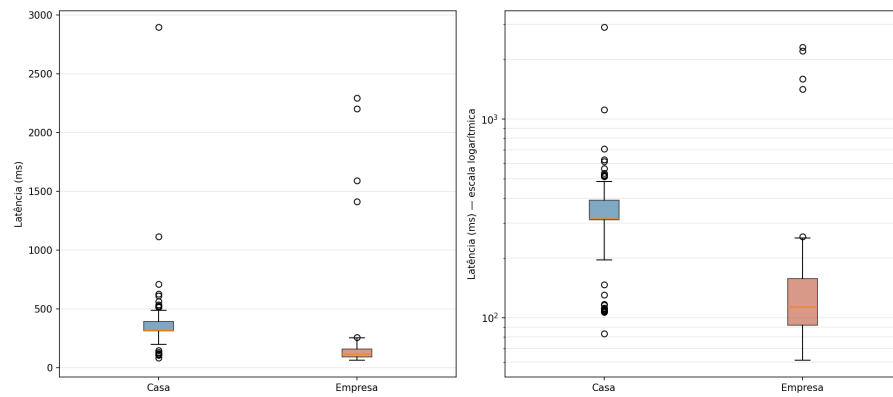


Figura 5.2: Comparação da latência de comunicação do comando de alteração de modo entre redes (*boxplot*).

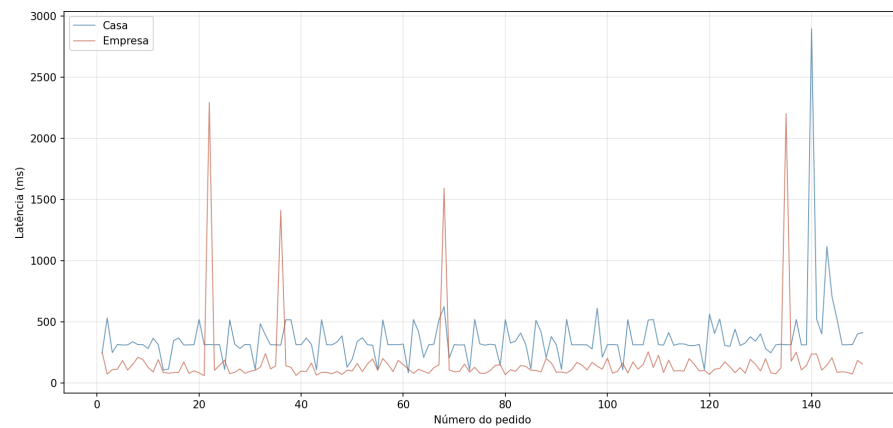


Figura 5.3: Evolução temporal da latência de comunicação ao longo das amostras recolhidas, por rede.

5.1.2 Tempo Total de Resposta

O segundo ensaio mede o tempo total entre o envio de um comando de posicionamento e a chegada efetiva do sistema à posição-alvo. Utilizou-se também um *script* responsável por enviar a sequência necessária para definir a posição pretendida, utilizando a variável `targetPos`. A lógica do PLC assume o controlo e executa o movimento até verificar a condição de posição definida na máquina de estados. O estado do sistema é monitorizado por *polling* do *endpoint* com cadência de 500 ms. Foram realizadas 50 amostras por rede, alternando entre movimentos de subida (0% → 25%) e de descida (25% → 0%), totalizando 25 amostras por sentido. Ao contrário do ensaio anterior, este caracteriza a totalidade da cadeia de resposta: comunicação, processamento e movimento. Os resultados são apresentados na Figura 5.4.

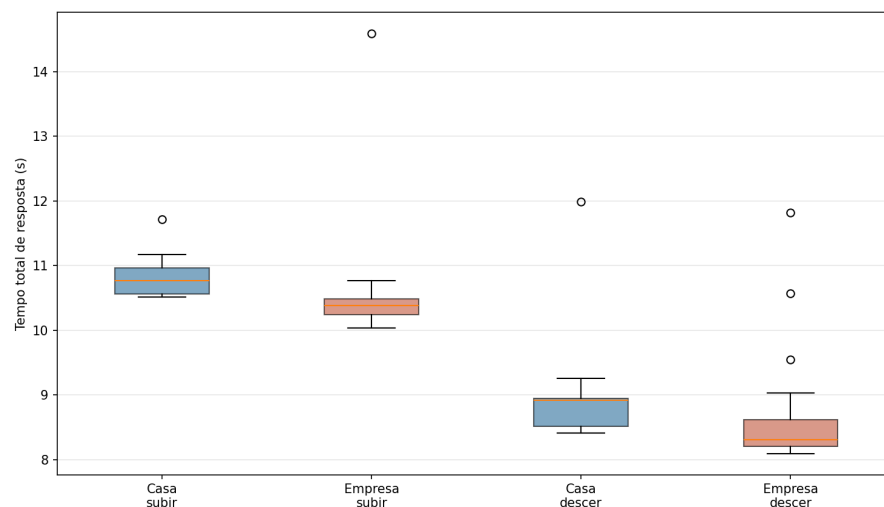


Figura 5.4: Comparação do tempo total de resposta ao comando de posicionamento entre redes, por sentido de movimento (*boxplot*).

É possível concluir que o ambiente de rede praticamente não afeta o tempo total de resposta, dado que os tempos são muito próximos entre as duas redes, apesar da diferença significativa de latência de comunicação observada no ensaio anterior. Este resultado confirma que o tempo total é dominado pelo movimento mecânico do atuador, tornando a diferença de latência de comunicação irrelevante para a experiência do utilizador.

5.1.3 Discussão de resultados

A análise conjunta dos dois ensaios permite extrair duas conclusões substantivas. A primeira é que o ambiente de rede afeta significativamente a latência de comunicação pura, mas, mesmo na pior das condições medidas, a latência mediana permanece bem abaixo do limiar de um segundo geralmente associado à percepção de manipulação direta em sistemas interativos. A escolha do REST sobre HTTP como protocolo de comunicação revela-se, assim, adequada ao caso de uso, mesmo em condições de rede menos favoráveis.

A segunda conclusão é que o tempo total de resposta do sistema é dominado pelo movimento e não pela comunicação. A diferença entre redes torna-se praticamente desprezável quando se mede o tempo até à chegada efetiva à posição-alvo. Esta observação tem uma implicação arquitetural relevante, uma vez que a otimização da camada de comunicação tem retorno limitado num sistema dominado por dinâmica mecânica lenta. Confirma-se também o desacoplamento entre a receção do comando e a sua execução, com o OPTA a responder rapidamente ao pedido HTTP e o movimento a decorrer de forma assíncrona, sem bloquear novos pedidos.

5.2 Caracterização dos tempos de curso do atuador

Com o intuito de calibrar a estimativa temporal de posição descrita no Capítulo 4 e de avaliar a estabilidade mecânica do AirLight ao longo do tempo, foi conduzido um conjunto de ensaios

para registrar os tempos de abertura e fecho do sistema. Foi desenvolvido um algoritmo auxiliar no OPTA, com o objetivo de comandar o sistema a realizar sucessivos ciclos de abertura e fecho até atingir cada um dos fins-de-curso, cronometrando os tempos totais de cada movimento durante todo o ensaio. Os ensaios foram distribuídos ao longo de 950 ciclos acumulados de operação, com nove medições realizadas em instantes distintos da vida do sistema. Os resultados são apresentados na Tabela 5.2.

Tabela 5.2: Tempos de abertura e fecho registados nos ensaios de caracterização.

Medição	Ciclos acumulados	Tempo de abertura (s)	Tempo de fecho (s)
1	1	38,03	32,69
2	46	38,26	32,71
3	412	37,68	32,97
4	459	38,04	32,97
5	528	37,90	32,86
6	573	38,02	32,75
7	597	37,62	32,45
8	950	38,01	33,14
9	985	38,42	32,91
Média	—	38,00	32,83

As medições revelam três aspetos relevantes para a validação do sistema de controlo. Em primeiro lugar, confirma-se uma diferença sistemática entre os dois sentidos de movimento: a descida é consistentemente mais rápida do que a subida, com uma diferença média de aproximadamente 5 segundos. Esta diferença é explicada pelo contributo da força gravítica no movimento descendente, que reduz a carga efetiva exigida ao atuador e aumenta a sua velocidade relativamente à subida, em que o motor tem de vencer integralmente o peso da estrutura. Esta assimetria é coerente com os resultados dos testes de tempo de resposta, onde a mesma diferença foi observada em movimentos comandados entre 0% e 25%, e com os incrementos diferenciados adotados na lógica de controlo (0,025 por ciclo a subir e 0,0312 por ciclo a descer), calibrados precisamente para refletir esta diferença de velocidade real entre sentidos.

Em segundo lugar, os tempos de curso apresentam elevada estabilidade ao longo dos 985 ciclos monitorizados. O desvio padrão é de 0,24 s para a abertura e de 0,20 s para o fecho, correspondendo a coeficientes de variação inferiores a 1%. Esta consistência indica que a velocidade do atuador não se degrada de forma perceptível ao longo da utilização, o que sustenta a viabilidade da estimativa temporal como método de controlo de posição para o horizonte de vida testado.

Em terceiro lugar, a estabilidade dos tempos de curso valida os incrementos de posição calculados no Capítulo 4. Se os tempos variassem significativamente ao longo do tempo, os incrementos fixos tornar-se-iam progressivamente inadequados e a estimativa divergiria da posição real independentemente da calibração periódica. A constância observada indica que os incrementos calibrados permanecem representativos do comportamento real dos atuadores.

Capítulo 6

Conclusões e Trabalho Futuro

O presente capítulo encerra a dissertação, sintetizando o trabalho desenvolvido e situando-o face aos objetivos inicialmente propostos. Começa por reunir as principais conclusões retiradas do desenvolvimento e da validação experimental do sistema, refletindo sobre os desvios entre o âmbito originalmente previsto e o trabalho efetivamente realizado. São também identificadas as direções de evolução futura, distinguindo entre aperfeiçoamentos à solução desenvolvida e oportunidades de integração com outros equipamentos e protocolos industriais.

6.1 Conclusões

O presente trabalho resultou no desenvolvimento de um sistema de controlo para o produto *AirLight*, abrangendo a programação do controlador industrial Arduino Opta em linguagens IEC 61131-3, o desenvolvimento de uma aplicação móvel como interface principal de operação, e a implementação de uma camada de supervisão experimental sobre a plataforma *Home Assistant*. A lógica de controlo implementada no Opta inclui uma máquina de estados para gestão dos modos de operação (manual, automático e emergência), um algoritmo de estimativa temporal de posição calibrado com base nos tempos de curso medidos experimentalmente, e funcionalidades de posicionamento por percentagem com tolerância definida. Foram, ainda, desenvolvidas duas rotinas complementares: uma de inicialização, que requer confirmação do operador sobre a posição física do equipamento no arranque, e uma de calibração automática, que corrige periodicamente a deriva da estimativa.

A validação experimental confirmou a adequação da arquitetura REST para o caso de uso. Os ensaios de latência mostraram valores medianos abaixo do limiar de resposta interativa em ambas as redes testadas, e os ensaios de tempo de resposta demonstraram que o tempo total é dominado pelo movimento mecânico, não pela comunicação. Os tempos de curso do atuador, medidos ao longo de 985 ciclos, apresentaram coeficientes de variação inferiores a 1%, sustentando a fiabilidade dos incrementos de posição adotados.

No domínio da segurança, foi implementada a integração com uma central de desenfumagem, com lógica *fail-safe* e estado de alarme dedicado na máquina de estados, assegurando a coexistência entre as funções de conforto e os requisitos de segurança contra incêndio. A capacidade de integração em redes industriais via Modbus TCP foi validada por prova de conceito, confirmando que o Opta pode expor variáveis a sistemas externos sem hardware adicional.

Importa enquadrar os resultados face à proposta de dissertação. A formulação original previa maior incidência na integração com equipamentos HVAC e sistemas BMS através de protocolos normalizados. Durante o desenvolvimento, a empresa identificou como prioritária a disponibilização de uma aplicação de controlo para utilizadores finais, o que levou a que a comunicação REST se tornasse o eixo principal do trabalho. A vertente de integração industrial foi mantida ao nível de prova de conceito e estudo prospetivo. Esta reorientação não comprometeu o objetivo de demonstrar a capacidade de integração do *AirLight* em ecossistemas digitais, tendo, contudo, resultado num trabalho com ênfase diferente da inicialmente prevista. Ainda assim, todos os objetivos definidos para a dissertação foram atingidos com sucesso: foi realizada a revisão do estado da arte, desenvolvida a lógica de controlo estruturada com suporte a múltiplos modos de operação, implementada a arquitetura de comunicação e as interfaces de utilizador, validado experimentalmente o desempenho do sistema, e estudada a integração com sistemas de supervisão e outros equipamentos do edifício, ainda que esta última vertente tenha sido validada ao nível de prova de conceito em vez de uma implementação completa em ambiente de produção.

6.2 Trabalho Futuro

As direções de trabalho futuro identificadas distinguem-se pelo seu âmbito. Por um lado, aperfeiçoamentos e extensões ao sistema desenvolvido, decorrentes de limitações ou oportunidades identificadas durante o próprio desenvolvimento. Por outro, oportunidades de integração do *AirLight* com outros equipamentos e protocolos industriais, que ampliariam o seu posicionamento em ecossistemas de automação predial mais amplos.

6.2.1 Consolidação do sistema

A quantificação rigorosa da precisão da estimativa de posição face à posição física real do *AirLight* não foi realizada. A principal razão está no facto do erro acumulado, ser de magnitude reduzida, então, qualquer medição manual da posição física, por exemplo com fita métrica, estaria ela própria sujeita a uma incerteza de medição da mesma ordem de grandeza do erro que se pretende quantificar, comprometendo a validade dos resultados. A abordagem que permitiria uma validação rigorosa e contínua da precisão seria a integração de sensores de posição absoluta, como encoders ou sensores de efeito Hall, que forneceriam feedback real em tempo de execução. Esta instrumentação, para além de permitir a validação, eliminaria a própria necessidade de estimativa temporal, constituindo uma evolução natural da arquitetura de controlo.

A substituição dos sensores simulados no BMS experimental por sensores físicos reais permitiria validar o comportamento das estratégias de supervisão em condições operacionais e quantificar o seu impacto energético efetivo, transitando de um ambiente de demonstração para um cenário representativo. As automações implementadas no BMS coexistem com priorização implícita. A implementação de um mecanismo explícito de resolução de conflitos, com hierarquia de prioridades documentada, aumentaria a robustez da camada de supervisão em cenários com múltiplas estratégias concorrentes. A funcionalidade de resposta automática ao sensor de temperatura do próprio *AirLight*, descrita no Capítulo 4, não considera atualmente se a abertura do sistema corresponde a uma estratégia de arrefecimento eficaz face às condições exteriores presentes. Distintamente da ventilação preventiva por calor previsto, que se baseia em temperaturas previstas, esta avaliação exigiria a medição em tempo real de ambas as temperaturas, interior e exterior, permitindo distinguir situações em que a abertura efetivamente contribui para o arrefecimento do espaço de situações em que a temperatura exterior seria igual ou superior à interior, tornando a ventilação ineficaz como estratégia de arrefecimento.

6.2.2 Integração industrial

O *Biocooler*, equipamento de arrefecimento evaporativo comercializado pela empresa, comunica por Modbus RTU sobre RS485, utilizando uma placa de comunicação dedicada que expõe parâmetros de controlo através de funções Modbus standard. O Opta dispõe de interface RS485 nativa, pelo que a integração é tecnicamente viável, atuando o Opta como *master* num barramento RS485 onde o *Biocooler* seria *slave*, permitindo coordenar ventilação natural e arrefecimento evaporativo. Esta integração não foi realizada, por o equipamento não estar disponível para testes e por a sua interface estar ainda a ser estudada internamente pela empresa.

Em contextos prediais onde BACnet ou KNX são os protocolos dominantes, a integração do *AirLight* pode ser concretizada através de *gateways* de protocolo, abordagem consagrada na indústria e discutida na revisão de literatura. Padrões emergentes como o BACnet/SC (*Secure Connect*), que aproxima os protocolos industriais das tecnologias baseadas em IP, podem simplificar esta integração no futuro. A segurança das comunicações Modbus TCP constitui também uma área relevante para implementações em ambiente operacional. A ausência de mecanismos nativos de autenticação ou cifra implica que a exposição do servidor Modbus do Opta a redes partilhadas exige medidas complementares, como segmentação de rede e políticas de *firewall*.

Bibliografia

- [1] Xiaoxin Wang, Chris Kendrick e Ray Ogden. «Influences of Lighting Control and Natural Ventilation on Energy Use and Overheating for a Day-lit Industrial Building». Em: *Proceedings of the Eleventh International IBPSA Conference*. 2009, pp. 1650–1654.
- [2] Jingxin Nie et al. «The effects of dynamic daylight-like light on the rhythm, cognition, and mood of irregular shift workers in closed environment». Em: (2021), 11:13059. DOI: [10.1038/s41598-021-92438-y](https://doi.org/10.1038/s41598-021-92438-y).
- [3] Ö. Sümengen e G. M. Şengönül. «Investigation of daylight performance in traditional residential buildings in the context of EN 17037 standard». Em: (2022), pp. 376–396.
- [4] Xue Li e Yupeng Wu. «A review of complex window-glazing systems for building energy saving and daylight comfort: Glazing technologies and their building performance prediction». Em: *Journal of Building Physics* 48.4 (2025). DOI: [10.1177/17442591241269182](https://doi.org/10.1177/17442591241269182).
- [5] Per Heiselberg. *Principles of Hybrid Ventilation*. 2002.
- [6] Zitao Jiang et al. «A literature review of cross ventilation in buildings». Em: *Energy and Buildings* 291 (2023), p. 113143. DOI: [10.1016/j.enbuild.2023.113143](https://doi.org/10.1016/j.enbuild.2023.113143).
- [7] Sara Omrani et al. *Effect of natural ventilation mode on thermal comfort and ventilation performance: Full-scale measurement*.
- [8] Dolaana Khovalyg et al. «Critical review of standards for indoor thermal environment and air quality». Em: *Energy and Buildings* 213 (2020), p. 109819. DOI: [10.1016/j.enbuild.2020.109819](https://doi.org/10.1016/j.enbuild.2020.109819).
- [9] Jared Landsman, Gail Brager e Mona Doctor-Pingel. «Performance, Prediction, Optimization, and User Behavior of Night Ventilation». Em: *Energy and Buildings* 166 (2018), pp. 60–72. DOI: [10.1016/j.enbuild.2018.01.026](https://doi.org/10.1016/j.enbuild.2018.01.026).
- [10] Nikolai Artmann, Heinrich Manz e Per Heiselberg. «Climatic potential for passive cooling of buildings by night-time ventilation in Europe». Em: *Applied Energy* 84.2 (2007), pp. 187–201. DOI: [10.1016/j.apenergy.2006.05.004](https://doi.org/10.1016/j.apenergy.2006.05.004).
- [11] «A framework for assessing the current and future capability of mechanical night ventilation in the context of climate change». Em: *Energy Reports* (2024). Study on current and future performance of night ventilation under climate change scenarios.

- [12] Jordan Correa et al. «Evolution of warm nights in the Canary Islands (1950–2023): evidence for climate change in the Southeastern North Atlantic». Em: *Theoretical and Applied Climatology* 156.87 (2025). DOI: [10.1007/s00704-024-05290-1](https://doi.org/10.1007/s00704-024-05290-1).
- [13] Thomas Schulze e Ursula Eicker. «Controlled natural ventilation for energy efficient buildings». Em: *Energy and Buildings* 56 (2013), pp. 221–232. DOI: [10.1016/j.enbuild.2012.09.024](https://doi.org/10.1016/j.enbuild.2012.09.024).
- [14] Wolfgang Kastner et al. «Communication Systems for Building Automation and Control». Em: *Proceedings of the IEEE* 93.6 (2005), pp. 1178–1203. DOI: [10.1109/JPROC.2005.849740](https://doi.org/10.1109/JPROC.2005.849740).
- [15] Gail Brager, Erik Ring e Kelly Powell. «Mixed-Mode Ventilation: HVAC Meets Mother Nature». Em: *Engineered Systems* (mai. de 2000). Available at eScholarship, University of California: <https://escholarship.org/uc/item/4014d8pd>, pp. 56–62.
- [16] Huyen Do e Kristen S. Cetin. «Mixed-Mode Ventilation in HVAC System for Energy and Economic Benefits in Residential Buildings». Em: *Energies* 15.12 (2022), p. 4429. DOI: [10.3390/en15124429](https://doi.org/10.3390/en15124429).
- [17] Daniel Plörer et al. «Control Strategies for Daylight and Artificial Lighting in Office Buildings — A Bibliometrically Assisted Review». Em: *Energies* 14.13 (2021), p. 3852. DOI: [10.3390/en14133852](https://doi.org/10.3390/en14133852).
- [18] C. Aghemo et al. «Management and Monitoring of Public Buildings through ICT Based Systems: Control Rules for Energy Saving with Lighting and HVAC Services». Em: *Frontiers of Architectural Research* 2 (2013), pp. 147–161. DOI: [10.1016/j.foar.2012.11.001](https://doi.org/10.1016/j.foar.2012.11.001).
- [19] Saman Taheri, Paniz Hosseini e Ali Razban. «Model predictive control of heating, ventilation, and air conditioning (HVAC) systems: A state-of-the-art review». Em: *Journal of Building Engineering* 60 (2022), p. 105067. DOI: [10.1016/j.jobe.2022.105067](https://doi.org/10.1016/j.jobe.2022.105067).
- [20] Zijing Tan e Xiang Deng. «An optimised window control strategy for naturally ventilated residential buildings in warm climates». Em: *Sustainable Cities and Society* 57 (2020), p. 102118. DOI: [10.1016/j.scs.2020.102118](https://doi.org/10.1016/j.scs.2020.102118).
- [21] Pooja Rajendra Varma, Bhagsen J. Parvat e V. K. Jadhav. «HVAC and Lighting Control Using PLC and SCADA for Integrated Building Management System». Em: *International Journal of Research in Engineering and Technology* 6.8 (2017), pp. 130–137.
- [22] Pedro Domingues et al. «Building automation systems: Concepts and technology review». Em: *Computer Standards & Interfaces* 45 (2016), pp. 1–12. DOI: [10.1016/j.csi.2015.11.005](https://doi.org/10.1016/j.csi.2015.11.005).
- [23] Dong-Hyun Kim et al. «Predictive Optimal Control of Indoor Temperature Using Deep Reinforcement Learning and Modbus TCP Communication». Em: *Applied Sciences* 15.13 (2025), p. 7248. DOI: [10.3390/app15137248](https://doi.org/10.3390/app15137248).

- [24] Wikipedia contributors. *Modbus*. Acedido em: 29 de julho de 2026. 2026. URL: <https://en.wikipedia.org/wiki/Modbus>.
- [25] Chrysi K. Metallidou, Kostas E. Psannis e Eugenia Alexandropoulou Egyptiadou. «Energy Efficiency in Smart Buildings: IoT Approaches». Em: *IEEE Access* 8 (2020), pp. 63679–63699. DOI: [10.1109/ACCESS.2020.2984461](https://doi.org/10.1109/ACCESS.2020.2984461).
- [26] Jacek Wyrębowicz, Krzysztof Cabaj e Jerzy Krawiec. «Messaging Protocols for IoT Systems — A Pragmatic Comparison». Em: *Sensors* 21.20 (2021), p. 6904. DOI: [10.3390/s21206904](https://doi.org/10.3390/s21206904).
- [27] Niccolò Aste, Massimiliano Manfren e Giorgia Marenzi. «Building Automation and Control Systems and performance optimization: A framework for analysis». Em: *Renewable and Sustainable Energy Reviews* 75 (2017), pp. 313–330. DOI: [10.1016/j.rser.2016.10.072](https://doi.org/10.1016/j.rser.2016.10.072).
- [28] VELUX. *Window or door comprising at least one openable lid part*. European Patent EP 2003262 A2. 2008.

Anexo A

Código AppOpta

O código desenvolvido no âmbito desta dissertação, associado à interface web e aplicação móvel desenvolvidas, está disponível no seguinte repositório:

<https://github.com/joaopmsilvaa/AppOpta>