



DOI: 10.31416/rsdv.v10i1.328

Aplicação Web para automação de ar-condicionado via integração das tecnologias Arduino e Power Line Communication

Web Application for air conditioning automation via integration of Arduino and Power Line Communication technologies

FARIAS, Ricardo Alves de. Graduação/Sistemas de Informação

Universidade Estadual do Sudoeste da Bahia (UESB) - Campus Jequié. Av. José Moreira Sobrinho, s/n, Jequezinho - Jequié - Bahia - Brasil. CEP: 45.200-000 / Telefone: (73) 3528-9723 / E-mail: ricardo14231@hotmail.com

PEIXOTO, Saulo Correa. Mestre/Ciência da Computação

Universidade Estadual do Sudoeste da Bahia (UESB) - Campus Jequié. Av. José Moreira Sobrinho, s/n, Jequezinho - Jequié - Bahia - Brasil. CEP: 45.200-000 / Telefone: (73) 3528-9723 / E-mail: saulopeixoto@hotmail.com

LOVATO, Agnaldo Volpe. Doutor/ Ciência da Computação

Universidade Estadual do Sudoeste da Bahia (UESB) - Campus Jequié. Av. José Moreira Sobrinho, s/n, Jequezinho - Jequié - Bahia - Brasil. CEP: 45.200-000 / Telefone: (73) 3528-9723 / E-mail: agnaldovl@uesb.edu.br

SANTOS, Alex Ferreira dos. Doutor/ Ciência da Computação

Universidade Federal do Recôncavo da Bahia (UFRB) - Campus Feira de Santana. Rua Godofredo Rebello de Figueiredo Filho (Antiga Avenida Centenário), nº 697, Bairro SIM - Feira de Santana - Bahia - Brasil. CEP: 44085-132 / Telefone: (75) 3622-9351 / E-mail: alex.ferreira@ufrb.edu.br

MANIÇOBA, Robson Hebraico Cipriano. Doutor/ Engenharia Elétrica

Universidade Estadual do Sudoeste da Bahia (UESB) - Campus Jequié. Av. José Moreira Sobrinho, s/n, Jequezinho - Jequié - Bahia - Brasil. CEP: 45.200-000 / Telefone: (73) 3528-9723 / E-mail: rhcmanicoba@uesb.edu.br

RESUMO

O artigo tem como objetivo apresentar o processo de desenvolvimento de um sistema de automação, para gestão de aparelhos/equipamentos de ar-condicionado, que integra a tecnologia PLC (Power Line Communication) e a plataforma Arduino. Para isso, este projeto foi implementado em três etapas. Na primeira, foi realizada a programação e montagem dos componentes à placa Arduino. Em seguida, foram efetuados os testes de integração do notebook à placa Arduino via PLC. Por último, foi desenvolvido o software de gerenciamento para interface entre o(a) operador(a) e o servidor Arduino. O software torna a interface mais amigável, e disponibiliza módulos que facilitam a gestão da automação dos ares-condicionados. A PLC foi escolhida por proporcionar transmissão de dados a longas distâncias, com a infraestrutura já existente (rede elétrica), para reduzir custos na implementação da solução. Como resultados, os testes apresentaram baixa latência e ausência de perda de pacotes na transmissão das informações, que resulta em um desempenho excelente para a comunicação entre os componentes por meio da tecnologia PLC. Além disso, o software de gerenciamento apresenta diversas informações relevantes dos ares-condicionados, e possibilita a(o) operador(a) um gerenciamento eficiente.

Palavras-chave: Automação, PLC, Ar-condicionado, Arduino.

ABSTRACT

The article aims to present the process of developing an automation system for the management of air conditioning devices/equipment, which integrates PLC (Power Line Communication) technology



and the Arduino platform. For this, this project was implemented in three stages. In the first one, the programming and assembly of the components to the Arduino board was carried out. Then, the integration tests of the notebook to the Arduino board via PLC were carried out. Finally, the management software was developed to interface between the operator and the Arduino server. The software makes the interface more user-friendly, and provides modules that facilitate the management of the automation of air conditioners. The PLC was chosen for providing data transmission over long distances, with the existing infrastructure (electrical network), to reduce costs in the implementation of the solution. As a result, the tests showed low latency and absence of packet loss in the transmission of information, which results in an excellent performance for communication between components through PLC technology. In addition, the management software presents a lot of relevant information from the air conditioners, and allows the operator to efficiently manage it.

keywords: Automation, PLC, Air conditioner, Arduino.

Introdução

A utilização consciente de energia elétrica é um dever de todos, uma vez que a projeção da demanda energética entre os anos de 2019 a 2023 no setor residencial e comércio é de 3,8% e 4% ao ano, respectivamente, segundo dados da Empresa de Pesquisa Energética (EPE) (BRASIL, 2019b). As principais fontes de geração de eletricidade no Brasil foram as usinas hidrelétricas, com 59% da eletricidade gerada, e as usinas termoeletricas, com 24,6% no ano de 2020 (BRASIL, 2021). Por consequência, o uso de termoeletricas causam um grande impacto ao meio ambiente por utilizar recursos não renováveis e emitir grande quantidade de poluentes na atmosfera, mesmo as hidrelétricas que utilizam recursos renováveis são nocivas ao ecossistema local (BONDARIK; PILATTI; HORST, 2018).

O consumo de energia elétrica pelos órgãos públicos custa cerca de 2 bilhões de reais por ano à União (BRASIL, 2019a), sendo de competência das instituições a responsabilidade no uso racional deste recurso. Desta maneira, faz-se necessário que as instituições possuam políticas e rotinas para o uso coerente dos recursos energéticos. Nesse sentido, uma das formas de conseguir esta eficiência é através da automação, tendo em vista que a inserção de procedimentos manuais corrobora para falhas no processo. Segundo Maitelli (2008) e Goeking (2010), a automação é o processo de substituir o trabalho braçal do homem por equipamentos programáveis que utilizam uma potência elétrica ou mecânica a fim de torná-lo mais eficiente, seguro e economicamente mais acessível.

Ao pensar em automatizar de forma eficiente os recursos elétricos, iniciamos um estudo na Universidade Estadual do Sudoeste da Bahia (UESB), campus Jequié. Esta possui o setor de Audiovisual, sendo uma de suas atribuições controlar o acionamento e temperatura dos ar-condicionados durante o período de aula. Entretanto, esse processo é realizado de forma manual, no início de cada aula, dependendo do colaborador(a) do setor de audiovisual deslocar-se até a sala para acionar o ar-condicionado. Este processo pode ocasionar a permanência do ar-condicionado em funcionamento sem a utilização do espaço físico em situações adversas, tais como: alteração repentina de sala, alteração do horário de aula, demora do(a) colaborador(a) em deslocar-se para desligar o ar-condicionado, dentre outros fatores. Além disso, existe alta demanda do(a) colaborador(a) em horários pontuais, visto que a universidade possui horários padronizados para o



início e término de cada aula, sendo que o(a) mesmo(a) tem que se deslocar em todas as salas que possuem aula naquele horário.

Desta forma, foi desenvolvido um sistema automatizado para leitura de dados da sala e controle do ar-condicionado, que busca contribuir no processo de gerenciar os ares-condicionados visando a economia de energia e almejando o princípio da eficiência na atividade de gestão. Os dados da sala compreendem a leitura dos sensores de temperatura e presença. Para a interface da automação com o usuário, foi desenvolvido o Software de gerenciamento da automação com foco no ambiente acadêmico. Com ele é possível selecionar um pavilhão e a sala de aula onde o dispositivo está alocado, além de visualizar os horários de aula cadastrados para a sala.

A automação do ar-condicionado foi realizada com a integração da placa Arduino acoplada à placa Ethernet shield para prover uma interface de rede e a tecnologia PLC. O fato de usar o cabeamento elétrico como meio de propagação dos dados de rede foi um fator determinante para a escolha da tecnologia PLC, como meio de comunicação entre os dispositivos de rede com a placa Arduino, visto que o uso da infraestrutura existente reduz consideravelmente os custos do projeto.

Por fim, foi desenvolvida uma ferramenta eficaz de gerenciamento de controle dos equipamentos de ares-condicionados para a UESB, com a realização da automação entre as tecnologias desse estudo.

As próximas seções deste artigo estarão organizadas da seguinte maneira: na seção 2 será abordado o estudo da arte. Na seção 3 será apresentada a fundamentação teórica. A seção 4 descreverá os materiais e métodos utilizados na pesquisa. A seção 5 apresentará os resultados e discussão. Por fim, na seção 6 estarão as considerações finais.

Estado da Arte

Nesta seção, serão apresentados os resultados de alguns trabalhos, identificados na literatura, que tratam da mesma temática abordada na pesquisa. Serão analisados como cada autor(a) abordou e aplicou as tecnologias na área de automação residencial/industrial e o meio de transmissão de dados utilizado.

Costa e Lima (2015) desenvolveram um sistema de automação e monitoramento do consumo de energia em uma residência. Em uma plataforma de ensaio, os autores utilizam o Arduino UNO, o shield de Ethernet e um módulo de relé para acionar as cargas que foram representadas por lâmpadas e um ponto de conexão para tomada elétrica. Foi utilizado o sensor ACS712 para identificar a passagem de corrente e assim realizar a aferição. O sistema se diferencia por prover uma aplicação com uma interface de fácil utilização, disponibiliza relatórios com o consumo dos equipamentos e possibilita a persistência dos dados usados pela aplicação, desenvolvido no banco de dados MySQL.

Corroborando com o estudo em questão, Moreira (2019) desenvolveu a integração entre o Arduino e a tecnologia PLC com o objetivo de ser aplicado para automação residencial e industrial. Na arquitetura do projeto o Arduino atua como servidor da aplicação. Ao receber requisições do navegador, o mesmo exibe uma página HTML (*Hypertext Markup Language*) com as opções para controle de dispositivos. O código da aplicação, bem como o código que coloca as portas digitais com



nível lógico alto/baixo para acionar o dispositivo conectado em sua interface, representado por LEDs (diodo emissor de luz) são armazenados na memória flash do próprio Arduino. A comunicação entre o cliente e o Arduino é realizado via PLC. Nesse estudo, os testes realizados obtiveram excelentes resultados com latência entre o servidor e o dispositivo na rede inferior a 1 milissegundo para as distâncias de 10m, 50m e 150m. O trabalho apresentado expõe a diversidade de aplicações possíveis com a integração da plataforma Arduino e o meio de transmissão de dados apresentado uma vez que os resultados foram satisfatórios e disponibiliza mais uma opção para a comunicação entre os equipamentos do sistema de automação.

Em Neto (2011) é projetado a automação do sistema de iluminação de uma residência controlado por um tablet iPad via rede Wi-Fi (*Wireless Fidelity*). O circuito do protótipo possui um relê ligado na porta digital do Arduino. O relê é responsável pelo acionamento ou interrupção da passagem de corrente elétrica para a lâmpada. No teste realizado o autor relata que ocorreu perda de pacotes de rede conforme a intensidade do sinal Wi-Fi diminuía, o que ocasionava falta de sincronismo entre a informação exibida no iPad e o estado da lâmpada acionada. Logo, quando isso acontecia, era enviado o comando para ligar a lâmpada, o status do iPad era alterado, mas o comando para acionamento da mesma não chegava até o Arduino. Desta forma, o problema apresentado pelo autor poderia ser resolvido adicionando uma função de *callback*, de forma básica, tal função enviaria para a aplicação a confirmação do acionamento do relê o que diminuiria a ocorrência de falta de sincronismo no sistema automatizado.

Outro estudo significativo foi proposto por Cardoso (2014), no qual foi apresentado um projeto de automação residencial entre a plataforma Arduino e um aparelho de telefonia móvel com comunicação entre as plataformas via rede de telecomunicação. Foi utilizado o shield SIM900 para prover a funcionalidade de envio de mensagens SMS (*Short Message Service*) ao Arduino. O autor montou um circuito com os sensores de luminosidade (*Light Dependent Resistor* - LDR) e o de corrente elétrica. Os dados obtidos dos sensores são enviados para o aparelho celular via SMS. Desta forma, a integração com a rede de telefonia possibilita o envio de comandos de forma remota ao sistema automatizado. Outra vantagem é a utilização do sensor LDR, uma vez que pequenas alterações no projeto possibilitam realizar a integração com o sistema de iluminação da residência e programar uma determinada faixa de luminosidade para acionar/desativar a iluminação do ambiente de forma autônoma.

Uma desvantagem no estudo de Cardoso (2014), é que o sistema pode apresentar maior demora de interação com os equipamentos automatizados e maior tempo para o *feedback* ao usuário, visto que o envio de SMS depende da intensidade do sinal da operadora e pode apresentar maior lentidão se comparado a outros meios de transmissão expostos nos projetos apresentados, como por exemplo, a rede Wi-Fi.

Fundamentação teórica

A automação surgiu do processo de evolução das técnicas do homem na busca por formas de simplificar o trabalho e auxiliar em suas atividades cotidianas. Para Silva et al. (2019), a automação



é a operação de um sistema com a mínima interferência do operador humano. Segundo Ribeiro, Ribeiro e Simon (2019) o conceito de automação varia conforme o contexto de sua aplicação, sua perspectiva é alterada conforme o indivíduo e sua experiência. Em outras palavras, para uma dona de casa, um operário industrial ou uma pessoa comum, a automação vai estar presente de diferentes formas, pode ser representada por uma máquina de lavar, um robô ou um caixa eletrônico, respectivamente.

Com o advento da Revolução Industrial, a mudança da produção artesanal para a produção em larga escala criou/gerou uma busca contínua por melhoria nas formas de produção a fim de manter a competitividade entre as organizações. Nesse período era predominante a automação de máquinas hidráulicas e pneumáticas. O desenvolvimento de microprocessadores na década de 70 inseriu novos componentes que mudaram a característica da automação (ROSARIO, 2009). Controladores lógicos programáveis e sistemas integrados passaram a fazer parte deste cenário, possibilitando a inserção de equipamentos com elevado poder de processamento, maior confiabilidade do sistema, capacidade de se comunicar com outros equipamentos, entre outros fatores (MAITELLI, 2008).

Para classificar a automação, em se tratando de ambientes domésticos, foi criado o termo domótica. Este é caracterizado pela integração de diferentes sistemas com instruções pré-programáveis que visam proporcionar as necessidades básicas de conforto, comunicação, segurança e gestão energética do ambiente residencial (MURATORI; DAL BÓ, 2011). Desta forma, para haver a interação entre o Software e o Hardware é utilizado o Arduino. Este é uma plataforma *open source* de desenvolvimento baseado em microcontroladores ATmega. A facilidade de desenvolver um projeto no Arduino, bem como sua abrangência em termos de aplicabilidade, possibilita sua utilização pela comunidade de desenvolvedores (ARDUINO.CC, 2021).

Por meio do Hardware e software, o Arduino proporciona um ambiente de fácil utilização, que através de instruções gravadas no microcontrolador torna possível ler dados de entrada (dados de um sensor ou pressionar de um botão) e gerar uma determinada saída (acionar uma lâmpada, motores etc). As instruções são geradas no ambiente de desenvolvimento do Arduino - IDE (*Integrated Development Environment*) usando uma linguagem de programação baseada em C/C++ (ARDUINO.CC, 2021).

A plataforma Arduino disponibiliza um conjunto de placas que se diferenciam pela quantidade de memória, processamento, quantidade de pinos de entrada/saída, entre outras características. Em especial, os dois modelos de placas mais utilizados são a UNO (Figura 1a) e a MEGA (Figura 1b). Além disso, é possível adicionar funcionalidades a placa através de Shields. Estas são placas de extensão que acopladas ao Arduino acrescentam uma determinada funcionalidade. Com isso, as Shields se tornam uma plataforma genérica aplicáveis em diferentes áreas e para diversos fins. Como exemplo, o Ethernet possibilita conectar o Arduino à rede e acessá-lo através de um navegador web, por sua vez, o motor Shield L293D permite ligar e controlar motores de corrente contínua ilustrados nas Figuras 2a e 2b, respectivamente (FILIPEFLOP, 2022).

Com as tecnologias supracitadas, temos que definir qual o meio utilizado para realizar a transmissão das informações. Dentre as tecnologias apresentadas, a PLC está sendo amplamente utilizada por diversos países (KOMINE; NAKAGAWA, 2003), (FERREIRA et al, 2011). É importante

ressaltar que esta tecnologia utiliza a rede elétrica como meio de transmissão da informação, ou seja, utiliza a infraestrutura já existente, o cabeamento elétrico das residências e indústrias. Desta forma, o custo é menor, em comparação com as demais. Para isso, o sinal da informação é modulado e transportado com o sinal senoidal de uma portadora via o cabeamento de rede elétrica. No receptor existem filtros e processadores para separar o sinal da informação da onda portadora (PAVLIDOU et al, 2003). Os sinais da rede elétrica e da informação podem coexistir no mesmo meio de transmissão, sem interferência entre si, devido as técnicas de modulação e a faixa de frequência que operam (CARCELLE, 2009). O órgão regulamentador de cada país define a faixa de frequência para o PLC. No Brasil, a Agência Nacional de Telecomunicações (Anatel) estabelece a faixa de 1,705MHz a 50MHz para o PLC de banda larga (ANATEL, 2009) a qual fica distante da faixa frequência de 60Hz utilizada pela rede elétrica.

Figura 1. Modelos de placa Arduino

a) UNO



b) Mega



Fonte: (ARDUINO.CC, 2021).

Figura 2. Placas Shields

a) W5100



b) L293.



Fonte: (ARDUINO.CC, 2021).

A topologia PLC é classificada em três níveis de acordo a sua abrangência, são elas: Outdoor, Indoor e Modem PLC. A extensão do primeiro corresponde a rede externa ao ambiente do receptor, ou seja, a rede de baixa tensão que liga o transformador ao quadro de distribuição da residência. O segundo abrange os circuitos internos da residência/prédio até os pontos de distribuição. Por último, o modem PLC que é responsável pela interface dos dispositivos com os pontos de distribuição (PAVLIDOU et al, 2003), (CARCELLE, 2009).



Materiais e métodos

O presente trabalho é classificado como uma pesquisa experimental. Segundo (GONÇALVES, 2020), este tipo de pesquisa consiste em determinar um objeto de estudo, selecionar as variáveis que seriam capazes de influenciá-lo e definir as formas de controle e de observação dos efeitos que a variável produz no objeto.

Deste modo, foi proposto como objeto a integração da plataforma Arduino, usada para automação dos ares-condicionados, com o sistema de gestão dos mesmos via tecnologia PLC. Buscou-se verificar a viabilidade de tal integração e sua funcionalidade na comunicação dos dispositivos.

Esta pesquisa é de natureza aplicada, uma vez que visa a construção de uma solução de automação via integração entre as tecnologias existentes. Como definido por Souza et al. (2013), parte de um conhecimento pré-existente e, através da pesquisa e/ou experiência prática, busca a produção de novos materiais, produtos e aparelhagens, novos processos, sistemas e serviços ou aperfeiçoamento de sistemas, processos já existentes.

O desenvolvimento deste trabalho foi segmentado em três etapas. A primeira foi realizada a programação e montagem dos componentes à placa Arduino MEGA. Nesse estágio foram montados dois circuitos independentes, um é responsável pela clonagem dos códigos do controle remoto do ar-condicionado e o outro pela coleta dos dados dos sensores PIR (Sensor de Presença) e DHT 22 (Sensor de Temperatura e Umidade), bem como o acionamento do ar-condicionado através do emissor IR. Em razão de utilizar uma única placa durante o trabalho, o código e circuito que foram usados para clonar as teclas do controle remoto foram sobrescritos pelo conjunto (código e circuito) do servidor Arduino. Em seguida foram efetuados os testes de integração do notebook à placa Arduino via PLC. Por último foi desenvolvido o Software de gerenciamento para interface entre o operador e o servidor Arduino.

Os materiais usados para realização do trabalho foram um notebook de configuração: sistema operacional Windows 10 Pro de 64bits, processador Intel Core i7-6700HQ de 2,60GHz, 12GB de memória RAM DDR 3, SSD de 240GB. O dispositivo é responsável por executar o Software de gerenciamento dos ares-condicionados, assim como o banco de dados usado pela aplicação. A placa Arduino MEGA, composto pelo microcontrolador Atmega328p, foi utilizado para acionar e realizar a leitura de dados dos sensores através de suas portas digitais. Acoplado a placa, o Ethernet Shield W5100 fornece a interface de rede RJ45 para comunicação da placa Arduino por meio dos protocolos TCP (*Transmission Control Protocol*) ou UDP (*User Datagram Protocol*). A interface de rede do W5100 foi conectada ao adaptador PLC TL-PA4010 que possibilita o tráfego de dados lógicos pela rede elétrica até o segundo adaptador PLC, ligado a interface do roteador Wi-Fi. Os demais materiais utilizados foram os sensores de temperatura (DHT 22), sensor de presença (PIR) - modelo DYP-ME003 e o Led emissor infravermelho (IR).

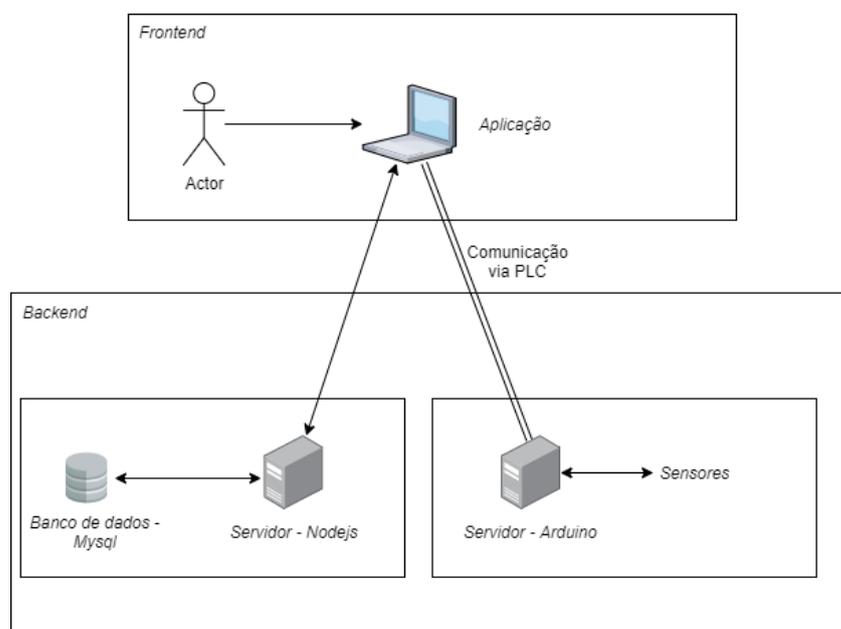
Nos próximos tópicos serão descritos a arquitetura do projeto, desenvolvimento do servidor Arduino, além do Software de gerenciamento e a integração entre ambos. O código da aplicação e do servidor Arduino desenvolvido nesse trabalho está disponível nos links do Apêndice A da monografia que foi realizado este trabalho (FARIAS, 2021), sob a licença *open source*, isto é, qualquer usuário pode modificar ou executá-lo.

Arquitetura

O Software de gerenciamento é composto por duas camadas: o *frontend* e *backend*. No desenvolvimento de sistemas Web o termo *frontend* é designado para representar a camada responsável pela interface direta com o usuário. Nessa camada está a aplicação de gerenciamento. Por sua vez o *backend* é responsável por todo o tratamento dos dados acessados pelo usuário e as ações realizadas, tais como, interação com o banco de dados, validação de segurança das informações, entre outros (ABDULLAH; ZEKI, 2014).

A aplicação acessa dois servidores. O primeiro encarregado por manipular as informações no banco de dados, que é o Software de gerenciamento e compreende os dados de cadastro com as informações do pavilhão, sala de aula, ar-condicionado, usuário e status de configuração atual do ar-condicionado. O segundo envia os comandos ao sensor IR. Os dados transmitidos entre a aplicação e o servidor Arduino é feita via PLC. A arquitetura do Software está representada na Figura 3.

Figura 3. Arquitetura do projeto.



Fonte: Elaborado pelos autores

Servidor Arduino

A plataforma Arduino é bastante popular na automação em razão do baixo custo, facilidade de desenvolvimento e diversas funcionalidades que podem ser acopladas à placa.

Para o estudo foi utilizado a placa Arduino MEGA em virtude da sua memória *flash* de 256KB, necessária para armazenar os códigos do controle do ar-condicionado. Outros quatro componentes foram utilizados para prover as funcionalidades de leitura dos dados do ambiente, indispensáveis para o projeto, sendo eles: sensor de temperatura, sensor de presença, emissor IR e o Ethernet shield.

Após a definição dos sensores, fez-se necessário fornecer uma interface de acesso aos dados via rede. Para isso, foi acoplado o Shield Ethernet W5100 que proporciona o acesso à camada de rede



do modelo OSI (*Open System Interconnection*), por conseguinte, possibilita à placa funcionar como servidor respondendo requisições HTTP (*Hypertext Transfer Protocol*), sendo o protocolo de Internet que permite a troca de dados entre um cliente e o servidor.

A estrutura do servidor foi desenvolvida no modelo de API (*Application Programming Interface*) que fornece acesso aos recursos a partir de um padrão de requisições predefinidas. As requisições do tipo GET para a URL (*Uniform Resource Locator*) `http://IP_SERVIDOR/?c=resSensores` retornam os dados dos sensores DHT 22 e PIR em notação JSON (*JavaScript Object Notation*), onde o IP_SERVIDOR deve ser substituído pelo IP atribuído ao Shield Ethernet.

Assim como o modelo de requisição anterior, o sensor IR é acionado sempre que é feita uma requisição do tipo GET com os parâmetros enviados na URL. Como exemplo, a URL `http://IP_SERVIDOR/?c=18&cool=true&fan=false` define a temperatura do ar-condicionado em 18 graus, ativa o modo frio e desativa o modo ventilar.

A Figura 4 inclui a configuração de atribuição das portas digitais da placa Arduino aos sensores, bem como, as bibliotecas que possuem métodos e definições utilizadas para facilitar o uso dos componentes.

Figura 4. Parâmetros de configuração dos sensores e Shield Ethernet.

```
1 //***** Bibliotecas utilizadas *****
2 #include <ArduinoJson.h>
3 #include <Ethernet.h>
4 #include <SPI.h>
5 #include <IRremote.h>
6 #include <DHT.h>
7
8 //***** Variáveis globais *****
9 char c;
10 String parametroURL;
11 boolean parametroReconhecido = false;
12 float temperatura;
13 int dadosPIR;
14 int tamanho = 60; //Tamanho do conjunto (blocos) do código RAW
15 int khz = 32; //Frequência do sinal em KHz
16 int presenca;
17
18 //***** Configuração sensor DHT *****
19 #define dhtPin 7 //Pino digital atribuído ao DHT22
20 #define dhtType DHT22 //Define o modelo do sensor (DHT22 / AM2302)
21 DHT dht(dhtPin, dhtType);
22
23 //***** Configuração sensor PIR (Presença) *****
24 #define pirPin 8 //Pino digital atribuído ao PIR
25
26 //***** Configuração emissor IR *****
27 int irPin = 9; //Pino digital atribuído ao IR
28 IRsend irSend;
29 IRrecv irRecv(irPin);
30
31 //***** Configuração do Shield ETHERNET *****
32 byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //Atribuição do endereço MAC ao shield W5100
33 byte ip[] = { 192, 168, 0, 130 }; //Atribuição do endereço IP da rede
34 byte gateway[] = {192, 168, 0, 1}; //Atribuição do Gateway da rede
35 byte subnet[] = {255, 255, 255, 0}; //Atribuição da máscara de rede
36
37 EthernetServer server(80);
38 EthernetClient client;
39
40 //***** Configuração do JSON *****
41 StaticJsonDocument<500> doc;
42 JsonObject json = doc.createNestedObject("sensores");
43
44 //***** Fim configurações *****
```

Fonte: Elaborado pelos autores



A implementação do servidor transcorreu em duas partes. A primeira foi a clonagem dos comandos do ar-condicionado. Foi utilizado o sensor receptor de infravermelho IR VS1838B para receber o sinal do controle remoto e o código IRrecvDump, disponível como código exemplo na biblioteca IRremote na versão 2.2.3, para decodificar e retornar as informações de acordo com o protocolo reconhecido.

Entretanto, podemos utilizar o método RAW para o envio dos comandos. A utilização desse método permite o envio de instruções sem a necessidade de especificar o protocolo utilizado pelo fabricante do ar-condicionado, isto é, um método generalista para envio do código da função que funciona na maioria dos aparelhos de ar-condicionado. Para isso, devemos obter o conjunto com o módulo dos números decimais que identificam o comando, tamanho do conjunto e a frequência utilizada pelo sinal. A biblioteca IRremote.h define a função sendRaw que recebe tais parâmetros.

Os testes foram realizados no ar-condicionado da marca LG modelo TSNC092TNW5. A Figura 5 sintetiza os passos para acionar o equipamento a partir do método RAW. A função loop executa o código entre as chaves em um ciclo contínuo, e, ao receber uma nova requisição, é feita a leitura dos caracteres da URL para identificar o comando. Nesse exemplo recebemos a ação para ligar o ar-condicionado. Uma vez identificado o comando o método sendRaw(liga, tamanho, khz), na linha 28, é enviado o sinal para o emissor IR.

Figura 5. Síntese do método SendRAW().

```
1 //***** Variáveis globais *****
2 char c;
3 String parametroURL;
4 boolean parametroReconhecido = false;
5 int tamanho = 60; //Tamanho do conjunto (blocos) do código RAW
6 int khz = 32; //Frequência do sinal em KHz
7
8 //***** Código RAW do controle remoto *****
9 unsigned int liga[60] = {8550, 3900, 800, 1450, 750, 350, 800, 300, 800, 300, 800, 1400, 800, 300, 750, 350, 750,
10 350, 750, 400, 650, 450, 650, 450, 650, 450, 650, 450, 650, 450, 650, 500, 550, 550, 600, 500, 600, 500, 550, 1650,
11 550, 1700, 500, 550, 600, 1650, 500, 550, 600, 550, 550, 550, 1650, 550, 1650, 550, 1650, 550};
12
13 //***** Função principal *****
14 void loop() {
15
16     client = server.available();
17
18     boolean fimLinha = true;
19     while (client.available()) {
20         c = client.read(); //Pega caracter por caracter (while)
21         parametroURL += c; //Concatena os caracter e forma o parametro
22
23         if (c == '\n' && fimLinha) { //Ao chegar no fim da URL de requisição, verifica quais comandos foram recebidos
24             .
25             .
26             .
27             if (parametroURL.indexOf("/?c=liga") > 0) {
28                 irSend.sendRaw(liga, tamanho, khz);
29                 delay(10);
30                 parametroReconhecido = true;
31                 resConfirmaAcao();
32             }
33             .
34             .
35             .
36 //***** Função para retornar a confirmação da requisição *****
37 void resConfirmaAcao() {
38     client.println(F("HTTP/1.0 200 OK"));
39     client.println(F("Content-Type: application/text"));
40     client.println("Access-Control-Allow-Origin: *");
41     client.println();
42 }
```

Fonte: Elaborado pelos autores



Software de gerenciamento

O software de gerenciamento foi desenvolvido com o objetivo de tornar a interface com o servidor Arduino mais amigável e disponibilizar módulos que facilitem a gestão da automação dos ares-condicionados da instituição. Para o desenvolvimento, foi utilizado o framework Angular (versão 10.1.1) e aplicado o método SPA (*Single Page Application*), de modo a permitir a navegação entre as páginas da aplicação sem a necessidade de o navegador recarregar a página a cada mudança de tela. O *backend* da aplicação, responsável por acessar o banco de dados, foi desenvolvido em NodeJs, versão 14.10.0. Para persistência dos dados, foi utilizado o banco de dados MySQL, versão 8.0.21.

Os dados persistidos são referentes aos parâmetros selecionados de configuração do ar-condicionado para cada sala. Por exemplo, quando o ar-condicionado é ligado via Software as informações de temperatura, status da função gelar e da função fan são armazenadas para quando for selecionado a visualização dos dados de outra sala. Na tela inicial da aplicação os parâmetros do ar-condicionado da primeira sala são armazenados para poderem ser recuperados posteriormente. Além da persistência dos parâmetros do ar-condicionado, também são armazenadas as informações necessárias para identificar o item de cada módulo, sendo elas:

- ✓ Módulo pavilhão: nome, quantidade de salas e pavilhão ativo.
- ✓ Módulo sala de aula: chave estrangeira do pavilhão, chave estrangeira do ar-condicionado, nome e sala ativa.
- ✓ Módulo horário: chave estrangeira da sala de aula, horário matutino, horário vespertino, horário noturno e horário ativo.
- ✓ Módulo ar-condicionado: nome, temperatura atual, status função gelar, status função fan, ligado, alocado, temperatura mínima do aparelho, temperatura máxima do aparelho, URL do dispositivo e ar-condicionado ativo.
- ✓ Módulo usuário: nome, login, senha, perfil e usuário ativo.

Além dos campos informados, para todos os itens são armazenados os campos: identificador, item ativo, item deletado, data de atualização e data de cadastro. Essas informações são necessárias para controle do registro na base de dados.

Resultados e Discussão

As telas de cadastro com os campos solicitados para cadastrar os itens de cada módulo podem ser visualizadas nas Figuras 6 a 10. Em cada módulo é possível realizar as operações de cadastrar, alterar, listar e deletar um item. As Figuras 11 a 15 ilustram a tela inicial de cada módulo, onde são listados os itens cadastrados. A Figura 16 exhibe o diagrama dos módulos desenvolvidos e a relação entre eles.

Figura 6. Cadastro - pavilhão.

AngularArduinoPlc x +
localhost:4200/home/Pavilion/new

ARDUINO PLC

Pavilhão Lista

Novo Pavilhão

Nome * Qtd. Sala *

Ativo

Cancelar Salvar

Fonte: Elaborado pelos autores

Figura 7. Cadastro - sala de aula.

AngularArduinoPlc x +
localhost:4200/home/Room/new

ARDUINO PLC

Sala Lista

Nova Sala

Pavilhão * Ar-condicionado *

Nome *

Ativo

Cancelar Salvar

Fonte: Elaborado pelos autores

Figura 8. Cadastro - ar-condicionado.

AngularArduinoPlc x +
localhost:4200/home/Ai/new

ARDUINO PLC

Ar-condicionado Lista

Novo ar-condicionado

Nome * Temp. Min * Temp. Max *

URL do dispositivo *

Ativo

Cancelar Salvar

Fonte: Elaborado pelos autores

Figura 9. Cadastro - horário.

Horário

Novo horário

Pavilhão * Sala *

Matutino Vespertino Noturno

Não informado

Hora	SEG	TER	QUA	QUI	SEX	SAB
07:30	-	-	-	-	-	-
08:10	-	-	-	-	-	-
09:00	-	-	-	-	-	-
10:00	-	-	-	-	-	-
10:50	-	-	-	-	-	-
11:40	-	-	-	-	-	-

Ativo

Cancelar Salvar

Fonte: Elaborado pelos autores

Figura 10. Cadastro - usuário.

Usuário

Novo Usuário

Nome *

Login * Senha * Confirmar senha *

Ativo

Cancelar Salvar

Fonte: Elaborado pelos autores

Figura 11. Tela inicial - Pavilhão.

Pavilhão

Lista

Id	Nome	Qtd. Salas	Ativo	Ações
1	Josélla Navarro	3	<input checked="" type="checkbox"/>	
2	Manoel Sarmento	14	<input checked="" type="checkbox"/>	
4	Novo Pavilhão	10	<input checked="" type="checkbox"/>	
5	Teste 5	15	<input type="checkbox"/>	
15	New Pavilion	10	<input type="checkbox"/>	

Items per page: 10 1 - 5 of 5

Fonte: Elaborado pelos autores

Figura 12. Tela inicial - Sala de aula.

Id	Sala	Ar-condicionado	Ativo	Ações
1	Sala 01	Ar SALA 01	<input checked="" type="checkbox"/>	
3	Sala 02	Ar SALA 02	<input checked="" type="checkbox"/>	
18	Sala 23	Ar SALA 07	<input type="checkbox"/>	

Fonte: Elaborado pelos autores

Figura 13. Tela inicial - ar-condicionado.

Id	Nome Ar	Alocação	Temp. min	Temp. max	URL do dispositivo	Ativo	Ações
1	Ar SALA 01	Sim	18	30	http://192.168.0.130/	<input checked="" type="checkbox"/>	
2	Ar SALA 02	Sim	18	30	http://192.168.0.130/	<input checked="" type="checkbox"/>	
7	Ar SALA 07	Não	18	30	http://192.168.0.130	<input checked="" type="checkbox"/>	

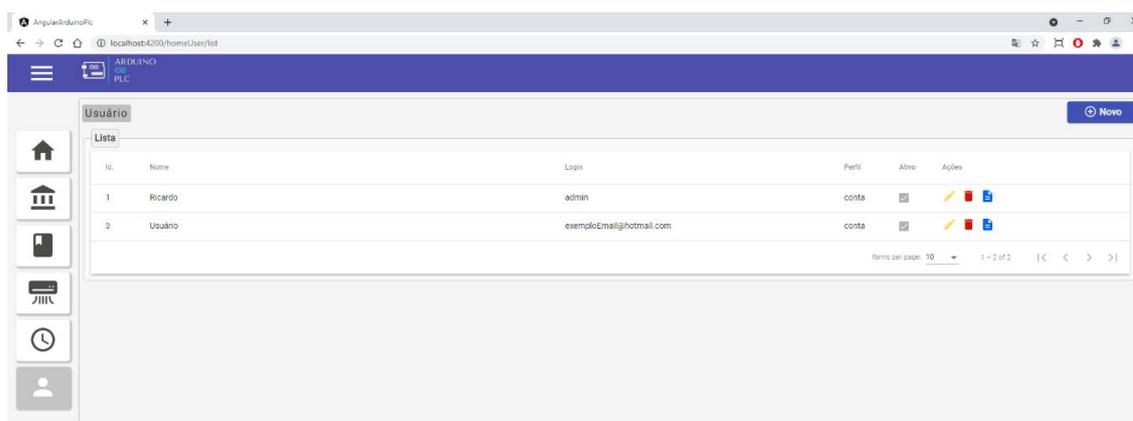
Fonte: Elaborado pelos autores

Figura 14. Tela inicial - horário.

Lista	Filtro	Sala	Turno	Filtrar																																										
<table border="1"> <thead> <tr> <th>Hora</th> <th>SEG</th> <th>TER</th> <th>QUA</th> <th>QUI</th> <th>SEX</th> <th>SAB</th> </tr> </thead> <tbody> <tr> <td>13:10</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> </tr> <tr> <td>14:00</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> </tr> <tr> <td>14:50</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> </tr> <tr> <td>15:50</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> </tr> <tr> <td>16:40</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> <td>AULA</td> </tr> </tbody> </table>	Hora	SEG	TER	QUA	QUI	SEX	SAB	13:10	AULA	AULA	AULA	AULA	AULA	AULA	14:00	AULA	AULA	AULA	AULA	AULA	AULA	14:50	AULA	AULA	AULA	AULA	AULA	AULA	15:50	AULA	AULA	AULA	AULA	AULA	AULA	16:40	AULA	AULA	AULA	AULA	AULA	AULA	Joséia Navarro	Sala 01	Vespertino	
Hora	SEG	TER	QUA	QUI	SEX	SAB																																								
13:10	AULA	AULA	AULA	AULA	AULA	AULA																																								
14:00	AULA	AULA	AULA	AULA	AULA	AULA																																								
14:50	AULA	AULA	AULA	AULA	AULA	AULA																																								
15:50	AULA	AULA	AULA	AULA	AULA	AULA																																								
16:40	AULA	AULA	AULA	AULA	AULA	AULA																																								

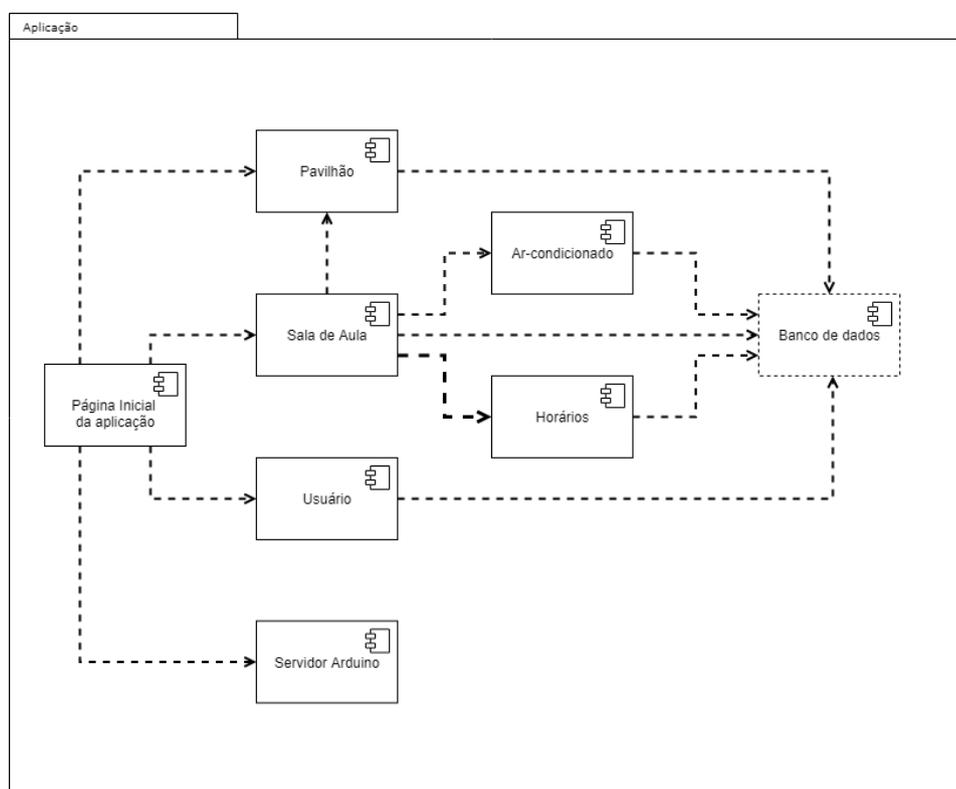
Fonte: Elaborado pelos autores

Figura 15. Tela inicial - usuário.



Fonte: Elaborado pelos autores

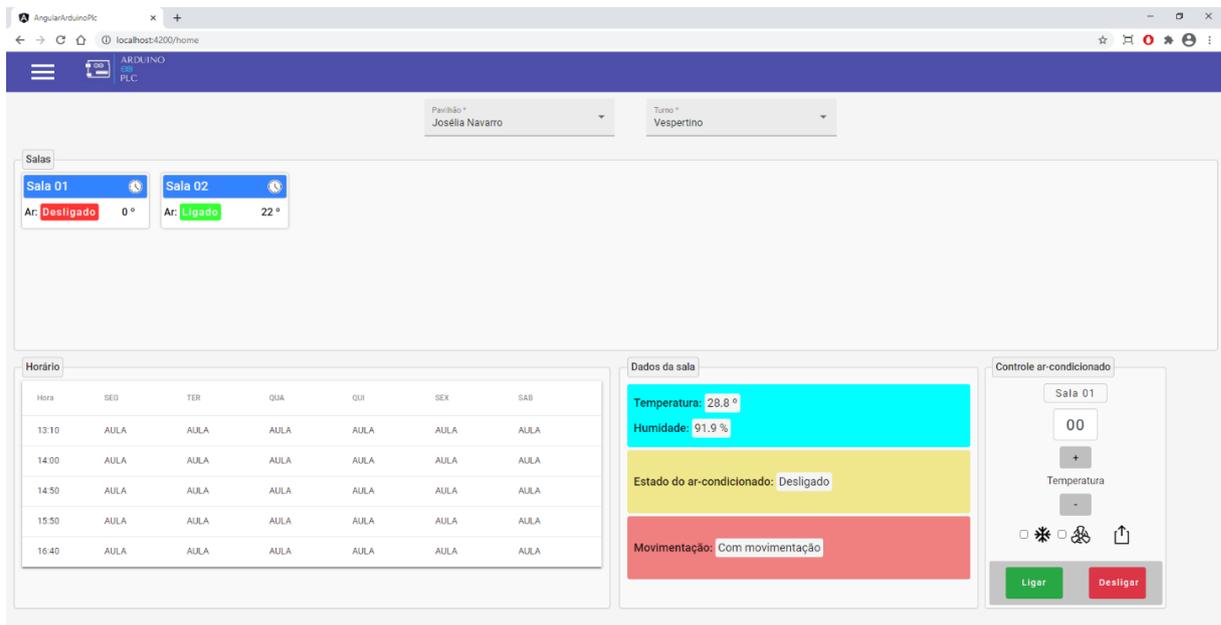
Figura 16. Diagrama de Componentes.



Fonte: Elaborado pelos autores

Na tela inicial é possível filtrar uma determinada sala para enviar uma requisição HTTP para o servidor Arduino, que por sua vez executa a ação e retorna a resposta da requisição. A aplicação exibe a resposta conforme o tipo de requisição feita. Assim, para requisições de dados dos sensores a aplicação agrupa as informações no elemento *Fieldset* da estrutura HTML denominado “Dados da Sala”, as requisições de acionamento do ar-condicionado exibem uma caixa de diálogo com a mensagem de êxito ou falha na operação. Além disso, é possível exibir os horários de aula cadastrada para a sala. A Figura 17 ilustra a tela inicial da aplicação.

Figura 17. Tela inicial da aplicação.



Fonte: Elaborado pelos autores

Finalmente, os demais módulos da aplicação são para fins de identificação do ar-condicionado, ou seja, selecionar o pavilhão e sala de aula em que o equipamento está alocado.

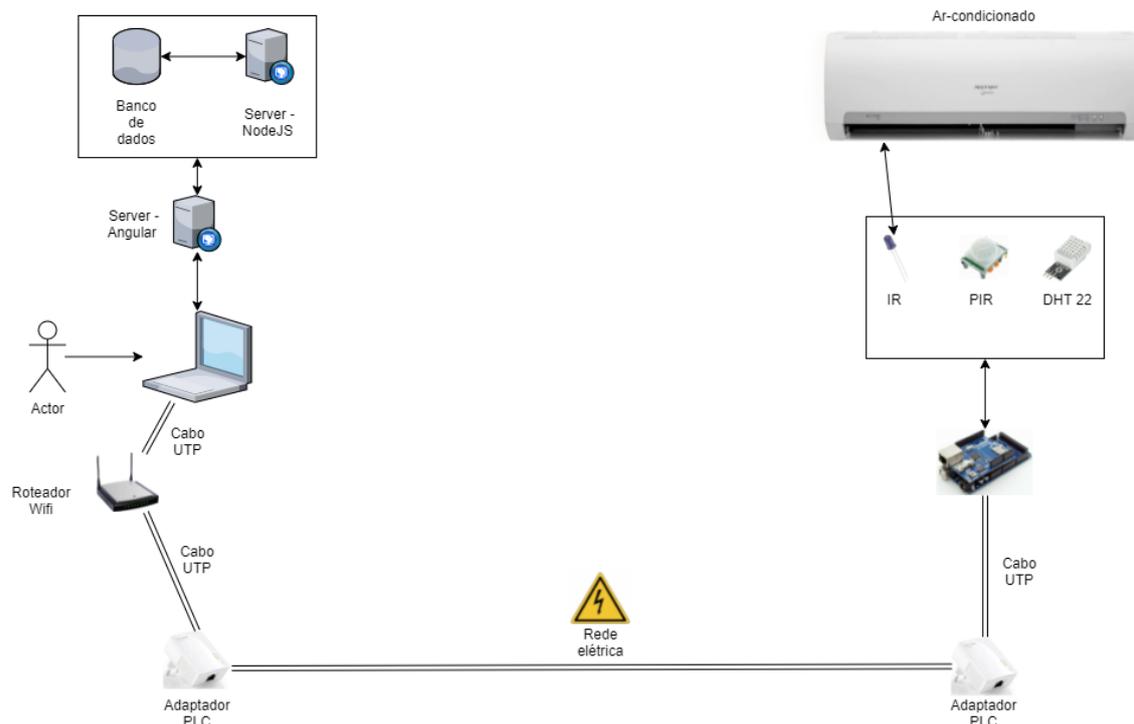
Integração realizada

Ao desenvolver o servidor Arduino, o software de gerenciamento realiza todo mapeamento de dados dos ares-condicionados. Para interligar a rede, estabelecendo a conexão através do cabeamento elétrico, um adaptador PLC foi utilizado na placa Arduino integrada ao aparelho de ar-condicionado e o segundo adaptador conectado à interface do roteador.

O adaptador PLC é capaz de enviar dados encriptados pela rede elétrica a uma taxa de transmissão de até 500Mbps, a uma distância máxima de 300 metros. O teste de comunicação realizado entre o cliente e o servidor Arduino se mostrou bastante estável, com baixa latência e sem perda de pacotes de rede.

Contudo, a distância entre os adaptadores PLC e a qualidade da estrutura do cabeamento elétrico podem afetar o desempenho obtido. As Figuras 18 e 19 ilustram o diagrama da integração entre o cliente e servidor e o resultado final da integração, respectivamente.

Figura 18. Diagrama do projeto.



Fonte: Elaborado pelos autores

Figura 19. Integração da aplicação com o servidor Arduino.



Fonte: Elaborado pelos autores

Conclusões

Executar um projeto de automação por vezes pode ser complexo, ter impacto monetário e demandar alterações/mudanças na estrutura do ambiente. Entretanto, a integração realizada neste trabalho demonstra a viabilidade do uso da infraestrutura elétrica existente como meio de comunicação entre os componentes da automação, por conseguinte, redução nesses efeitos.

O PLC proporciona transmissão de dados a uma distância de até 300m, o que é superior a outros meios comumente usados para transmitir dados de rede, como por exemplo o cabo de par



trançado em que o sinal é atenuado a distâncias superiores a 100m do emissor. Associado a isto, a plataforma Arduino disponibiliza fácil usabilidade na programação de microcontroladores.

Por fim, os testes realizados no trabalho apresentaram baixa latência e ausência de perda de pacotes, o que resulta em um desempenho excelente para a comunicação entre os componentes por meio da tecnologia PLC. Dessa maneira, a junção das tecnologias apresentadas proporciona novas possibilidades em projetos de automação com redução do custo e facilidade de execução do projeto.

Como sugestão para trabalhos futuros, pode-se implementar uma rotina para deixar o acionamento do ar-condicionado autônomo. No início de cada aula o ar-condicionado será ligado, sem a intervenção do(a) operador(a) do sistema, com a configuração de temperatura preestabelecida. Outras melhorias podem ser desenvolvidas, como a criação de um módulo que faça a leitura de todos os sensores de um laboratório, em intervalo de tempo predefinido. Desta forma, caso a temperatura atinja um determinado valor, será enviada uma mensagem via aplicativo para o responsável pelo laboratório. Tal funcionalidade visa evitar a perda de materiais sensíveis a temperatura que possam estar armazenados no local.

Referências

- ABDULLAH, H. M.; ZEKI, A. M. **Frontend and backend web technologies in social networking sites: Facebook as an example**. 3rd IEEE international conference on advanced computer science applications and technologies, 2014. p. 85-89, 2014.
- ANATEL. **Resolução no 527, de 8 de abril de 2009**. Disponível em: <<https://informacoes.anatel.gov.br/legislacao/resolucoes/2009/101-resolucao-527>>. Acesso em: 07 de Janeiro 2022.
- ARDUINO.CC. **What is Arduino**. Disponível em: <<https://www.arduino.cc/en/Guide/Introduction>>. Acesso em: 07 de Janeiro 2022.
- BONDARIK, R.; PILATTI, L. A.; HORST, D. J. Uma visão geral sobre o potencial de geração de energias renováveis no Brasil. **Interciencia**, v. 43, n. 10, p. 680-688, 2018.
- BRASIL. Ministério da Economia. **Portal de eficiência do Gasto ajudará órgãos federais a reduzir despesas com energia elétrica**, novembro 2019a. Disponível em: <<https://www.gov.br/economia/pt-br/assuntos/noticias/2019/11/portal-de-eficiencia-do-gasto-ajudara-orgaos-federais-a-reduzir-despesas-com-energia-eletrica>>. Acesso em 14 de dezembro de 2021.
- BRASIL. Ministério de Minas e Energia (MME), Empresa de Pesquisa Energia (EPE). **1ª Revisão Quadrimestral das Projeções da demanda de energia elétrica do Sistema Interligado Nacional 2019-2023**, Rio de Janeiro, abril de 2019b. Disponível em: <https://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes/PublicacoesArquivos/publicacao-305/topico-442/NT%201%C2%AA%20Revis%C3%A3o%20Quadrimestral%202019_FINAL.PDF>. Acesso em 14 de dezembro de 2021.
- BRASIL. Ministério de Minas e Energia (MME), Empresa de Pesquisa Energia (EPE). **Anuário Estatístico de Energia Elétrica 2021 ano base 2020**, p. 59, 2021. Disponível em <https://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes/PublicacoesArquivos/publicacao-160/topico-168/Anu%C3%A1rio_2021.pdf>. Acesso em 14 de dezembro de 2021.



- CARCELLE, X. **Power line communications in practice**. Artech House, 2009.
- CARDOSO, L. F. C. Sistema de automação residencial via rede celular usando microcontroladores e sensores. **Revista de Engenharia da Universidade Católica de Petrópolis**, v. 8, n. 2, p. 68-83, 2014.
- COSTA, A. A. F. DA; LIMA, P. R. B. D. Automação residencial com foco no consumo consciente de energia elétrica. **Revista Técnico-Científica**, v. 19, n. 34, p. 19-31, 2015.
- FARIAS, R. A. **Aplicação web para automação de ar-condicionado via integração das tecnologias arduino e power line communication**. Monografia (Bacharelado em Sistemas de Informação), Universidade Estadual do Sudoeste da Bahia (UESB), 2021.
- FERREIRA, H.C. et al. **Power line communications: theory and applications for narrowband and broadband communications over power lines**. John Wiley & Sons, 2011.
- FILIFELOP. Shields Arduino. Disponível em: <<https://www.filipeflop.com/categoria/arduino/Shields-arduino/>>. Acesso em: 07 de Janeiro 2022.
- GOEKING, W. Da máquina a vapor aos softwares de automação. **Portal O setor elétrico**, Santa Cecília, SP, 2010.
- GONÇALVES, J. R. **Manual de Projeto de Pesquisa**. Portal de Livros Abertos da Editora Processus, v. 12, n. 12, p. 01-67, 2020.
- KOMINE, T.; NAKAGAWA, M. Integrated system of white LED visible-light communication and power-line communication. **IEEE Transactions on Consumer Electronics**, v. 49, n. 1, p. 71-79, 2003.
- MAITELLI, A. L. Controladores lógicos programáveis. Universidade Federal do Rio Grande do Norte, Natal-RN, 2008. Disponível em: <<https://www.dca.ufrn.br/~maitelli/FTP/clp/>>. Acesso em: 07 de Janeiro 2022.
- MOREIRA, C. R. **Integração das tecnologias Arduino e Power Line Communication para automação residencial ou comercial via rede elétrica**. Trabalho de conclusão de curso de graduação, Universidade Estadual do Sudoeste da Bahia (UESB), Campus Jequié, 2019.
- MURATORI, J. R.; DAL BÓ, P. H. **Capítulo I Automação residencial: histórico, definições e conceitos**. O Setor elétrico, p. 70-77, 2011.
- NETO, R. O. **Automação de iluminação residencial utilizando microcontrolador Arduino e Tablet iPad via Wi-Fi**. Monografia (Bacharelado em Engenharia da Computação)-Centro Universitário de Brasília (UniCEUB, DF), 2011.
- PAVLIDOU, N. et al. Power line communications: state of the art and future trends. **IEEE Communications magazine**, v. 41, n. 4, p. 34-40, 2003.
- RIBEIRO, E. R. D.; RIBEIRO, L. G.; SIMON, M. Z. S. **A Internet das Coisas: Possibilidades Múltiplas de Conectividade**. Significações e estratégias midiáticas, p. 359, 2019.
- ROSARIO, J. M. **Automação industrial**. Editora Baraúna, 2009.
- SILVA, E. S. et al. Utilização da Computação Embarcada em um Shopping de Salvador. **Revista Computação Aplicada-UNG-Ser**, v. 5, n. 1, p. 26-37, 2019.
- SOUZA, D. I. et al. **Manual de orientações para projetos de pesquisa**. Novo Hamburgo: FESLVC, 2013.