



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO - UFERSA
DEPARTAMENTO DE ENGENHARIAS -
CENTRO MULTIDISCIPLINAR DE CARAÚBAS - CMC
BACHARELADO EM ENGENHARIA ELÉTRICA

JORGE HENRIQUE SOARES DE ALMEIDA

UTILIZAÇÃO DA INTERNET OF THINGS PARA SISTEMA DE
CADASTRO E MONITORAMENTO DE DADOS

CARAÚBAS - RN

2021



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO - UFERSA
DEPARTAMENTO DE ENGENHARIAS -
CENTRO MULTIDISCIPLINAR DE CARAÚBAS - CMC
BACHARELADO EM ENGENHARIA ELÉTRICA

JORGE HENRIQUE SOARES DE ALMEIDA

UTILIZAÇÃO DA INTERNET OF THINGS PARA SISTEMA DE CADASTRO E MONITORAMENTO DE DADOS

Trabalho de Conclusão de Curso submetido
à Universidade Federal Rural do Semi-Árido
como parte dos requisitos necessários para a ob-
tenção do Título de Engenheiro Eletricista.

Orientador: Tânia Luna Laura

CARAÚBAS - RN
2021

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

SA447 Soares de Almeida, Jorge Henrique.
u Utilização da Internet Of Things Para Sistema
de Cadastro e Monitoramento de Dados / Jorge
Henrique Soares de Almeida. - 2021.
46 f. : il.

Orientadora: Tania Luna Laura.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Engenharia
Elétrica, 2021.

1. Internet of Things. 2. ESP32. 3. MySQL. I.
Luna Laura, Tania, orient. II. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade
com AACR2 e os dados fornecidos pelo(a) autor(a).
Biblioteca Campus Caraúbas
Bibliotecária: Dalvanira Brito Rodrigues
CRB: 15/700

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

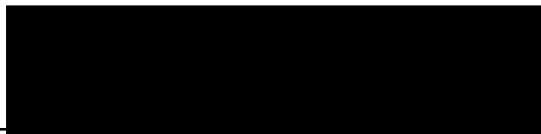
JORGE HENRIQUE SOARES DE ALMEIDA

**UTILIZAÇÃO DA INTERNET OF THINGS PARA
SISTEMA DE CADASTRO E MONITORAMENTO DE
DADOS**

Trabalho de Conclusão de Curso submetido
à Universidade Federal Rural do Semi-Árido
como parte dos requisitos necessários para a ob-
tenção do Título de Engenheiro Eletricista.

Defendida em: 19 de Novembro de 2021

Banca Examinadora



Tânia Luna Laura
Profa. Dra. (UFERSA)
Orientador



Francisco Luis de Carvalho Costa
Membro da Banca



Juan Rafael Filgueira Guerra
Prof. Me. (UFERSA)
Membro da Banca

CARAÚBAS - RN
2021

AGRADECIMENTOS

Dedico este trabalho primeiramente a Deus pela vida, pela força e a esperança que me concede a cada dia, a meus pais e meus amigos que sempre estiveram comigo me ajudando e apoiando em todas minhas decisões. A eles devo por me tornar uma pessoa capaz de lutar pelos meus sonhos e objetivos. Por fim, agradeço aos meus professores e também a minha orientadora pelos ensinamentos e apoio que ajudaram no desenvolvimento do presente trabalho.

"Conhecimento não é aquilo que você sabe, mas
o que você faz com aquilo que você sabe."
(Aldous Huxley)

RESUMO

A presença da internet hoje na vida das pessoas é algo essencial não só no que tange a comunicação e o entretenimento como também na vida profissional. A sua dependência é algo real e a cada dia se intensifica mais, com mais pessoas utilizando e acrescentando constantemente informações. Com sua evolução surgiram diversas aplicações com interação entre sensores, atuadores e dispositivos conectados à rede. Este tipo de interconexão é conhecida como Internet das Coisas (IoT). Muitas podem ser as aplicações da IoT, podendo ser desde algo mais simples e pessoal como a automação residencial a até uma aplicação mais profissional como o monitoramento de dados de algum processo industrial. De fato, com o surgimento dessa forma de interação através da internet, vários processos foram simplificados e vem se tornando cada vez mais efetivos. Neste trabalho será apresentada uma visão geral sobre a Internet das coisas, sua importância, perspectivas e exemplo de aplicação através do desenvolvimento de uma sistema de coleta, busca e inserção de dados a partir da conexão entre um dispositivo ESP32, uma interface gráfica desenvolvida e um banco de dados. Por fim, o trabalho apresentou os resultados esperados com o correto funcionamento do sistema de coleta e envio dos dados e da interface desenvolvida.

Palavras-chave: Internet of Things. ESP32. MySQL.

ABSTRACT

The presence of the internet in people's lives today is essential not only in terms of communication and entertainment, but also in professional life. Your dependence is real and it is intensifying every day, with more people constantly using and adding information. With its evolution, several applications emerged with interaction between sensors, actuators and devices connected to the network. This type of interconnection is known as the Internet of Things (IoT). There can be many IoT applications, ranging from something simpler and more personal like home automation to a more professional application like monitoring data from an industrial process. In fact, with the emergence of this form of interaction through the internet, several processes have been simplified and are becoming more and more effective. This work will present an overview of the Internet of Things, its importance, perspectives and application example through the development of a data collection, search and insertion system based on the connection between an ESP32 device, a developed graphical interface and a database. Finally, the work presented the expected results with the correct functioning of the data collection and sending system and the developed interface.

Keywords: Internet of Things. ESP32. MySQL.

LISTA DE ILUSTRAÇÕES

Figura 1 – Pinout ESP32	15
Figura 2 – Diagrama de bloco do ESP32	16
Figura 3 – LM35	17
Figura 4 – Esquema de ligação	18
Figura 5 – LDR	19
Figura 6 – DHT11 e sua ligação.	19
Figura 7 – DHT11 e sua ligação.	20
Figura 8 – Arquitetura do projeto	22
Figura 9 – Diferença das curvas AD	23
Figura 10 – Configurações do banco de dados	24
Figura 11 – Interface desenvolvida	24
Figura 12 – Projeto hospedado	25
Figura 13 – Correção AD no LM35	26
Figura 14 – Circuito do projeto	27
Figura 15 – Circuito real do projeto	27
Figura 16 – Comparação entre valores do DHT11 e LM35.	28
Figura 17 – Bibliotecas e definições utilizada.	28
Figura 18 – Comandos PHP e instância de objeto	29
Figura 19 – Conexões com o WiFi e DB	30
Figura 20 – Coleta e envio dos dados	30
Figura 21 – Dados coletados ID: 1 e 2	31
Figura 22 – Gráfico da Temperatura, Umidade e Luminosidade (ID 1)	32
Figura 23 – Gráfico da Temperatura, Umidade e Luminosidade (ID 4)	33
Figura 24 – Tabela da Temperatura, Umidade e Luminosidade (ID 5)	33
Figura 25 – Tabela da Luminosidade (ID 4)	34
Figura 26 – Gráfico da Luminosidade (ID 5)	34
Figura 27 – Enviando dados ao banco pela página HTML	35
Figura 28 – Enviando dados ao banco pela página HTML	35
Figura 29 – Consulta ID 7	36
Figura 30 – Valores finais (ID 5)	36

LISTA DE ABREVIATURAS E SIGLAS

IoT	<i>Internet of Things</i>
SQL	<i>Structured Query Language</i>
HTML	<i>HyperText Markup Language</i>
CSS	<i>Cascading Style Sheets</i>
PHP	<i>Hypertext Preprocessor</i>
ITU	<i>International Telecommunication Union</i>
ABINC	<i>Associação Brasileira de Internet das Coisas</i>
DB	<i>Database</i>
WiFi	<i>Wireless Fidelity</i>
ROM	<i>Read-only Memory</i>
RAM	<i>Random Access Memory</i>
IO	<i>Input Output</i>
AD	<i>Analog-Digital</i>
LDR	<i>Light Dependent Resistor</i>

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Justificativa	12
1.2	Objetivos	12
1.2.1	Objetivo Geral	12
1.2.2	Objetivos Específicos	12
2	REFERENCIAIS TEÓRICOS	13
2.1	Evolução da IoT	13
2.2	Banco de dados: conceitos e definições	14
2.3	ESP32	15
2.4	Sensores	17
2.4.1	LM35	17
2.4.2	LDR	18
2.4.3	DHT11	19
3	METODOLOGIA	21
3.1	Organização do projeto	21
3.2	Programação do ESP32	22
3.3	Configuração do banco de dados	23
3.4	Desenvolvimento da interface	24
3.5	Configurações para o acesso externo	24
4	RESULTADOS E DISCUSSÕES	26
4.1	Montagem e programação do ESP32	26
4.2	Coleta dos dados	31
4.3	CADASTRO E CONSULTA PELA PÁGINA CRIADA	33
5	CONCLUSÃO	38
	REFERÊNCIAS	39
6	APÊNDICES	41

1 INTRODUÇÃO

Há alguns séculos muito se temia com o processo de evolução tecnológica que acontecia, havia a preocupação da gradativa substituição da mão de obra humana por máquinas que eram mais rápidas, baratas e eficientes em certos processos. Contudo, a medida que se desenvolvia, novas atividades e trabalhos que não existiam começaram a aparecer e serem desenvolvidos, hoje o que existe é uma integração entre o homem e a máquina.

Quando falamos em invenções tecnológicas algumas coisas vêm à mente de forma quase que instantânea, seja elas mais antigas como a roda e a pólvora como também mais recentes como o motor a vapor e a internet. Sem dúvidas a internet foi um marco histórico que mudou completamente a vida de muitas pessoas, hoje é algo indispensável e presente na vida da grande maioria das pessoas no mundo.

Com ela foi possível mudar a forma com a qual as pessoas se divertem, se comunicam e trabalham. Atualmente ela é utilizada na chamada Internet of things (IoT), que pode ser definida, segundo Patel (2016) como uma rede que conecta qualquer coisa à Internet tendo como base protocolos estipulados através de equipamentos de detecção de informações para conduzir a troca de informações e comunicações a fim de obter reconhecimentos, posicionamento, rastreamento, monitoramento e decisões inteligentes.

Internet das coisas que vem do inglês *Internet of Things* (IoT) refere-se a objetos que podem ser legíveis, localizáveis, endereçáveis e podem agregar informações de outros objetos, realizando uma comunicação, são sem dúvida uma revolução da internet. Os objetos são reconhecíveis e possuem autonomia para tomar decisões ou dependendo da situação, permitir também o acesso a informações fornecidas por outros elementos e estabelecer uma comunicação (SINTEF e FRIESS, 2013).

Nos dias atuais, é difícil imaginar onde o processo de coleta, armazenamento e consulta de dados não esteja presente. No comércio e na indústria essa atenção se intensifica já que existem diversas informações que precisam ser monitoradas para um adequado funcionamento das atividades. Com base nisso, toda e qualquer forma de executar e aprimorar este processo se torna importante para inúmeras atividades.

No presente trabalho foi desenvolvido uma plataforma de monitoramento de temperatura, umidade e luminosidade, onde essas variáveis são coletadas por um microcontrolador ESP-32 e são armazenadas em um banco de dados MySQL. Para o monitoramento remoto, uma interface de usuário foi desenvolvida utilizando linguagens HTML, CSS e PHP. Por meio dessa interface é possível realizar consultas ao banco de dados e inserir valores, sendo possível o acompanhamento do comportamento dos dados ao longo do processo e a manipulação dos mesmos.

1.1 Justificativa

Integrar diversos processos tem se tornado mais necessário a cada dia, com isso, essas formas de integração acabam sendo mais requisitadas em diversos segmentos desde os mais simples como as atividades cotidianas que exercemos na rua ou em casa, até os mais complexos. Com base nessa necessidade de ampliação de comunicação das coisas, novos métodos e estudos sobre o tema acabam se tornando essenciais para o desenvolvimento dessas atividades. Por isso, este trabalho se apresenta como uma alternativa para essa questão, apresentando de forma didática, uma maneira simples de implementar um sistema de monitoramento com base na tecnologia IoT a partir da utilização de equipamentos de controle e de sensoramento relativamente baratos, de forma que o mesmo possa ser economicamente viável e consequentemente acessível uma grande parte da população.

1.2 Objetivos

1.2.1 Objetivo Geral

Aprofundar o conhecimento relacionado a IoT, com apresentação de seus métodos de aplicação através de dispositivos de processamento, interfaces gráficas e bancos de dados para armazenamento e utilização de informações, criando dessa forma um sistema de coleta, monitoramento e gestão de dados afim de difundir a ideia da implementação em sistemas e processos que possam ser otimizados a partir da tecnologia, promovendo agilidade, segurança e efetividade.

1.2.2 Objetivos Específicos

- Realizar experiências de coleta e armazenamento de dados utilizando uma placa de desenvolvimento conhecido como ESP32.
- Elaborar uma interface gráfica de usuário para facilitar o acesso a visualização de dados.
- Fazer a análise dos resultados obtidos a partir de algumas coletas e apontar a viabilidade do projeto realizado.

2 REFERENCIAIS TEÓRICOS

De acordo com Ashton (2009), a primeira vez que o termo foi utilizado foi em seu próprio trabalho “I made at Procter & Gamble” em 1999, 10 anos antes. O termo foi utilizado referenciando a tecnologia de RFID. Singer (2012) acredita que um outro possível surgimento do termo tenha vindo do artigo "When Things Start to Think" publicado também em 1999 por Neil Gershenfeld e que fazia uma abordagem sobre objetos que processavam informações.

Para Peter Waher (2015), IoT pode ser definida como a conexão entre algo e a internet de forma que estas não sejam operadas por pessoas. A principal forma de comunicação da internet existente hoje é a humana, contudo a IoT poderá se tornar uma nova vertente capaz de realizar aprendizagem Machine to Machine (máquina a máquina), garantindo uma maior conexão entre tudo.

De acordo com relatório da ITU (2005), é possível definir o termo como todo e qualquer dispositivo/objeto que possuam sensores, receptores e transmissores que possibilite e facilite a comunicação entre pessoas e objetos bem como entre objetos com outros objetos. Também foi definido pela ITU (2012) como “Uma infraestrutura global para a sociedade da informação, permitindo serviços avançados através da interconexão (física e virtual) de coisas baseadas em tecnologias interoperáveis de informação e comunicação, existentes e em evolução”.

A ideia geral da tecnologia é desenvolver um ecossistema de plataformas para objetos inteligentes que podem se conectar, de forma que a geração futura de dispositivos, softwares, interfaces e tecnologias de rede sejam integradas e possa ser difundida na sociedade para serem utilizadas em casa, na rua ou no trabalho (SINTEF e FRIESS, 2013).

2.1 Evolução da IoT

O primeiro eletrodoméstico ‘inteligente’ que surgiu, segundo Singer (2012), foi uma geladeira lançada pela LG em junho de 2000 na Coreia do Sul. A geladeira fazia parte de outros dispositivos que também se comunicavam com a internet sob um sistema da própria marca. Foi dito por Simon Kang, presidente da LG, que as pessoas poderiam usar a internet da geladeira como um rádio, uma tv, para uma aplicação web, um quadro de anotação, calendário e câmera digital.

Em 2005, houve a ascensão da IoT como assunto relacionado a segurança de dados e a privacidade, além de ser pautado pela ITU. Com o surgimento da banda larga, a organização passou a considera-la como “próximo passo da tecnologias ‘always on’ [...] que prometem um mundo de dispositivos interconectados em rede”. (Singer, 2012. Apoud. ITU, 2005).

Em 2008, houve a primeira conferência sobre o tema na Suíça, na cidade de Zurique, em seguida nos anos de 2010 e 2012 foram feitas outras conferencias internacionais em Tóquio

e na China. No Brasil houve uma conferência sediada em Salvador em 2010, com o tema: 1º Congresso de Tecnologia, Sistemas e Serviços com RFID.

De acordo com a ABINC (2018), seu mercado a nível global "alcançará US\$ 318 bilhões até 2023, a uma taxa composta de crescimento anual de 20%." Com grande parte destas tecnologias destinadas a serviços públicos e manufatura, chegando a bater mais de 50% do valor.

O aumento dessa tecnologia possibilitam o surgimento de novas ideias que talvez não estejam tão distantes como se possa imaginar, em Singer (2012), a autora apresenta algumas ideias das aplicações da tecnologia como em carros autônomos, fogão que prepara a comida preferida de alguém, telefones que fazem pedidos em restaurantes entre outras.

Ao se pesquisar sobre IoT no google, podem ser encontrados mais de 23.000 trabalhos sobre o tema e seus mais diversos tipos de aplicações com todos eles tendo sido publicados no ano de 2021.

2.2 Banco de dados: conceitos e definições

De acordo com os autores Silberschatz e Korth (2006), os banco de dados podem ser definidos como um sistema que coleta dados interrelacionados e de programas que são utilizados para a consulta e a manipulação dos dados, proporcionando um ambiente para o armazenamento e consulta. De acordo com a Oracle (2021), bancos de dados podem ser considerados uma “coleção organizada de informações - ou dados - estruturadas, normalmente armazenadas eletronicamente em um sistema de computador”. São controlados normalmente por um sistema de gerenciamento de banco de dados que junto a alguns aplicativos são chamados de sistema de banco de dados.

Um dos bancos de dados mais conhecido é o MySQL que é bastante utilizado no mundo devido a sua característica de distribuição e licença ser livre e de código aberto. É um servidor SQL de característica robusta, rápido e multiusuários (MYSQL, 2011). Surgiu com o intuito de garantir um sistema de armazenamento e compartilhamento livre onde os usuários podiam compartilhar e também alterar o produto. (LIMA, 2003).

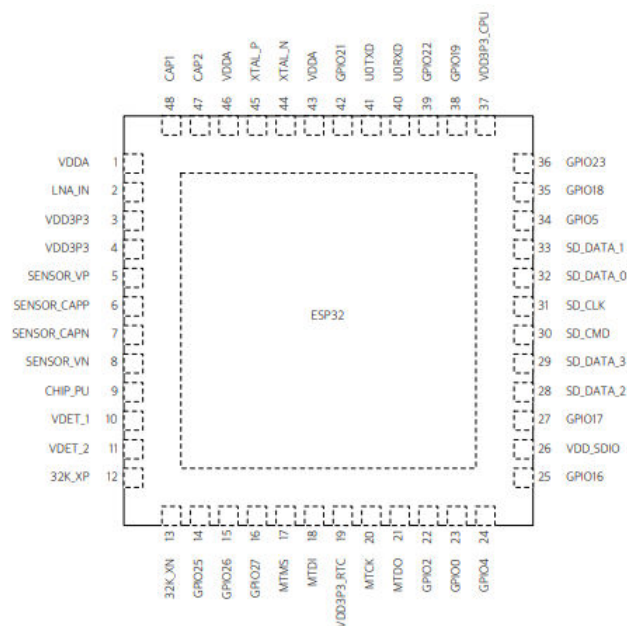
A sigla SQL significa Structured Query Language ou Linguagem de consulta estruturada. Ela foi desenvolvida em 1970 pela IBM junto com a Oracle e atualmente é uma linguagem de programação utilizada por quase todos os tipos de bancos de dados para diversas funções, tais como a consulta, inserção, a manipulação e algumas definições de dados.

Existem várias linguagens que podem ser utilizadas com o MySQL, como a C, C++, Java, PHP, Python, Pearl, Ruby entre outras. As plataformas também são bem diversificadas, como exemplo podem ser citadas o Windows, Linux, Sun Solaris, Apple MacOS X, entre algumas outras (MYSQL, 2011). Entre os usuários de MySQL, podem ser citados a Nokia, NASA, Dataprev, Banco Bradesco, HP, Friendster, Sony e Alcatel (SOUSA, 2008).

2.3 ESP32

Lançado em 2016 pela Espressif Systems, o microcontrolador ESP32 possibilita uma conexão com a internet através do WiFi, possui um consumo baixo, grande capacidade de processamento, conexão por Bluetooth, 448 KB de ROM, 520 KB de RAM e 36 portas IO com uma faixa de operação de 3,3V (SYSTEMS, 2019). Na Figura 1 é possível ver seu esquema de pinagens que conta com GPIOs, pinos ADC e DAC, UART e PWM.

Figura 1 – Pinout ESP32



Fonte: ESP32 Datasheet, Espressif Systems 2016

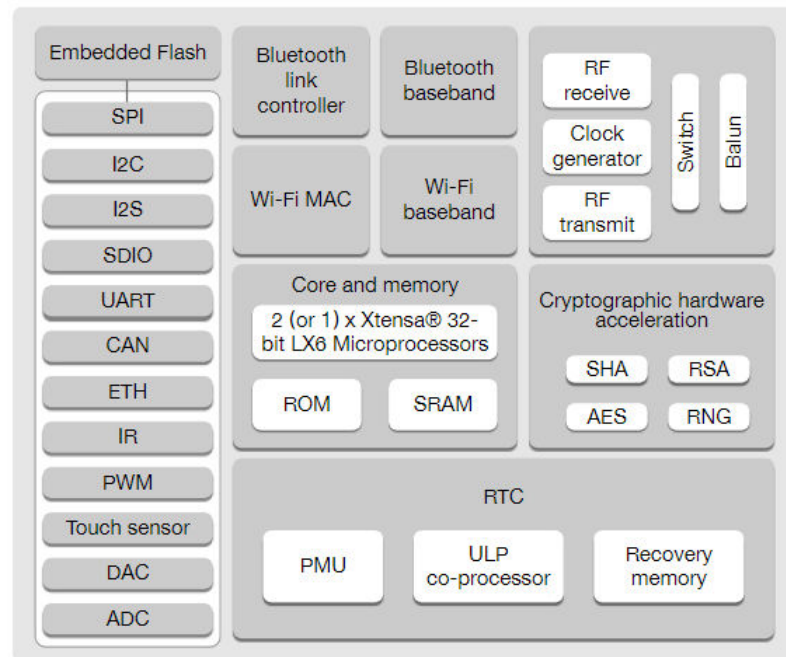
O ESP32 possui um notável processador que pode ser single ou dual-core e pode chegar a operar com até 240 MHz de clock, o que é um grande avanço ao se comparar com alguns microcontroladores conhecidos como o Arduino. Devido a isso e também a possibilidade do uso da internet e do bluetooth para conexões, rapidamente se tornou bastante conhecido e utilizado no mundo inteiro.

Outro ponto de destaque que facilita ainda mais a utilização do ESP32 é sua programação, que assim como o Arduino, também é programado através das linguagens C e C++ que são utilizadas no Software development kit que é fornecido pela sua desenvolvedora. Para quem já utiliza a IDE do Arduino, também é possível utilizá-la na programação do ESP32 através de uma biblioteca desenvolvida pela própria espressif (SANTOS, 2019). Essa possibilidade facilitou o seu uso, pois quem já programava o Arduino não teve muitas dificuldades em programar o ESP. Além dessa ser uma forma bastante comum de programa-lo, também é possível utilizar o Espressif IoT Development Framework e também o MicroPython.

Mais características sobre o dispositivo podem ser vistas na Figura 2, onde mostra alguns embarcados como os PWM's, sensor de temperatura, sensor touch, UART e o I2C, também é

visto suas memórias, transmissores e emissores RF e o bloco de RTC.

Figura 2 – Diagrama de bloco do ESP32



Fonte: Datasheet ESP32 (ESPRESSIF)

Abaixo é apresentada uma tabela comparativa entre o dispositivo ESP-32 e o Arduino uno, nela é possível ver alguns pontos onde o ESP-32 se destaca. Todos os dados foram retirados de suas respectivas folhas de dados.

Tabela 1 – Comparação ESP-32 e Arduino Uno

	ESP-32	Arduino Uno
Clock	160 MHz	16 MHz
Arquitetura	32 bits	8 bits
Cores	2	1
Bluetooth	Sim	Não
WiFi	Sim	Não
RAM	512 Kb	2 Kb
Flash	16 Mb	32 Kb
GPIO	32	14
ADC	18	6
DAC	2	0

Fonte: Autor

2.4 Sensores

Sensores são dispositivos capazes de responder a estímulos, sejam eles físicos ou químicos, através de sinais que podem ser medidos e monitorados. Podem ser sensíveis a formas de energias do ambiente, tais como luminosa, térmica, cinética, entre outras. Normalmente o sinal de saída nem sempre é suficiente para uso em sistemas de controle, sendo necessário realizar manipulações dos mesmos antes de sua leitura.

Os sensores podem ser analógicos ou digitais, sendo o primeiro caracterizado por poder assumir qualquer valor em seu sinal de saída ao longo do tempo desde que esteja dentro de sua faixa de trabalho. Já o segundo pode assumir apenas dois valores no sinal de saída ao longo do tempo, representados por 1 e 0 (Thomazini e Albuquerque, 2009).

Nessa seção serão abordados os tipos de sensores que foram utilizados para o desenvolvimento do projeto. Como a coleta feita foi de variáveis ambientais como são a temperatura, luminosidade e umidade, os sensores escolhidos foram o LM35, o LDR e o DHT11.

2.4.1 LM35

Este sensor se mostra bastante interessante de ser utilizado por algumas características que possui, como seu tamanho que é idêntico ao de um transistor, possibilitando sua aplicação em locais pouco acessíveis. Além disso o LM35 é considerado um sensor de precisão que capta temperaturas em uma faixa que vai de -55°C até 150°C com uma pequena variação de $0,25^{\circ}\text{C}$ até $0,75^{\circ}\text{C}$. Na Figura 2 é possível ver seu formato com um encapsulamento igual a de um transistor.

Figura 3 – LM35



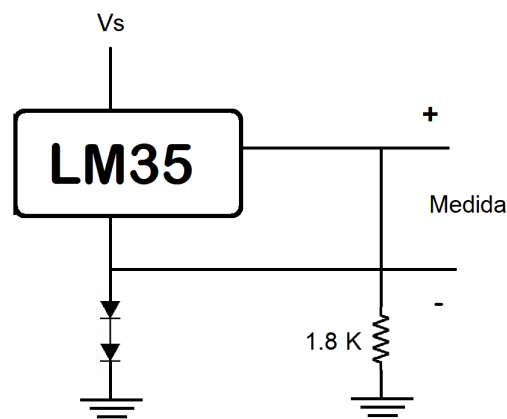
Fonte: Filipflop, 2021

Algumas de suas características são o baixo consumo que deve chegar ao máximo a

60 μ A, o fator de escala de 10mV para cada grau Celsius e sua tensão de alimentação que vai de 4V a 30V.

A ligação básica do LM35 consiste em 2 pinos para a alimentação em +Vcc e GND e o terceiro para o fornecimento do valor referente a temperatura medida. O tipo de ligação básica trabalha com valores em torno de 2°C até 150°C e será abordada mais a frente. Para trabalhar com seu range máximo é necessário uma configuração especial demonstrada na Figura 4.

Figura 4 – Esquema de ligação



Fonte: Autor

Com a presença dos diodos em série, a tensão no GND do LM35 passará de 0V a 1,4V, o que garante que conversores analógicos possam ler os valores referentes as tensões negativas já que alguns dispositivos com leitura AD não funcionam com valores negativos de tensão como é o caso de microcontroladores. Como não havia uma necessidade de medição de valores negativos de temperatura, não foi necessária a configuração.

2.4.2 LDR

O resistor dependente de luz, mais conhecido pela sigla em inglês LDR é um sensor que detecta variações de sinal luminoso. Devido a esta sua característica, podem ser utilizados para diversas aplicações, como em alarmes, sensor de presença e automatização com base na iluminação. Um exemplo de aplicação é na iluminação pública.

Assim como o LM35, o LDR é um sensor bastante discreto que pode ser implementado facilmente em qualquer dispositivo. Sua resistência varia entre 10 K Ω a 1 M Ω , com a menor resistência obtida na presença de luz e o valor máximo na sua ausência. De modo geral apresenta uma ampla possibilidade de aplicações, é bastante simples de se utilizar, boa durabilidade e bastante acessível.

Figura 5 – LDR



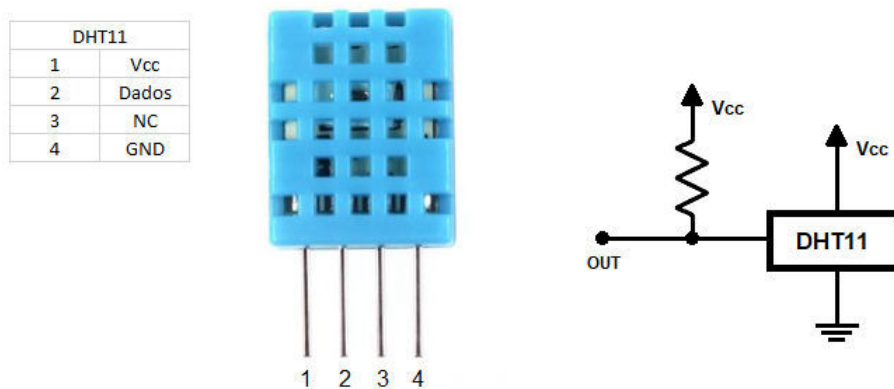
Fonte: Adrobotica, 2021.

2.4.3 DHT11

Este sensor possui um sistema de captação de temperatura e humidade que apresenta sua leitura em forma de uma sinal digital, o que permite uma maior confiabilidade e estabilidade com o tempo. O dispositivo é dotado de 4 pinos, porém em sua configuração são utilizados apenas três, dois para alimentação Vcc e GND e o outro para o sinal. Na Figura 6 é apresentado a imagem do sensor DHT11 juntamente com as identificações de seus pinos e a forma utilizada em sua ligação.

Algumas de suas características de funcionamento retiradas do seu próprio *datasheet* são: sua leitura de temperatura que possibilita leituras na faixa de 0 C° à 50 C°, a umidade relativa podendo ir de 20% até 90%, e suas respectivas precisões em 25 C° que são de $\pm 5\%$ e $\pm 2^\circ\text{C}$. Quanto a sua alimentação, ele é alimentado com uma tensão de 3VDC até 5VDC, podendo suportar até um máximo de 5,5VDC e a corrente por volta dos 200 μA a 500mA. Seu tempo de resposta é em torno de 2 segundos.

Figura 6 – DHT11 e sua ligação.



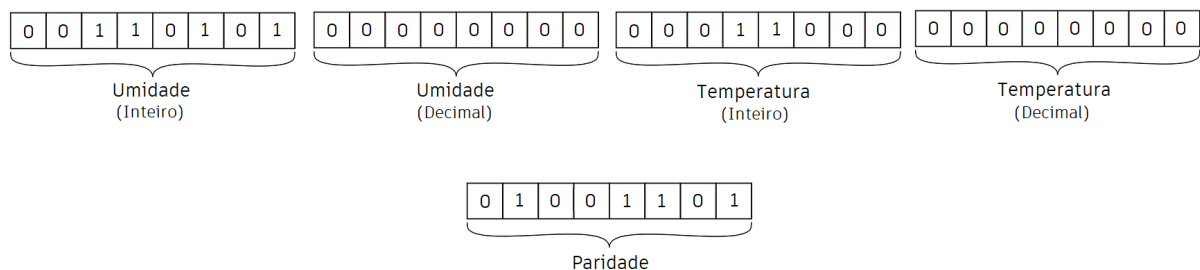
Fonte: Adaptado de: Datasheet DHT11 (ASAIR)

O funcionamento do DHT11 apesar de ser o mais complexo dos sensores até então apresentados, é bastante simples e consiste de um barramento que enviará as informações de

temperatura e umidade em um conjunto de oito bits. Funciona como um sistema mestre-escravo, onde o escravo enviará a informação a partir da solicitação do mestre. Sendo solicitado os dados do sensor, o DHT11 realizará uma transmissão de 40 dados, onde são divididos em 5, com 16 bits representando o valor inteiro de umidade e temperatura, outros 16 para os decimais dessas variáveis e mais 8 bits de paridade. Caso a soma dos 32 bits de umidade e temperatura seja diferente do byte de paridade, o recebimento é dado como incorreto e é solicitado um reenvio das informações.

A Figura 7 apresenta a forma de organização da informação que é utilizada pelo sensor com os 40 bits organizados em 5 bytes que descrevem o valor medido.

Figura 7 – DHT11 e sua ligação.



Fonte: Autor

3 METODOLOGIA

Este trabalho tem por objetivo demonstrar um tipo de aplicação de IoT baseado na coleta e armazenamento de dados com a possibilidade de manipulação através da inserção manual e com consultas através de uma interface gráfica simples e intuitiva. O sistema de monitoramento de dados utiliza a tecnologia MySQL para guardar as informações coletadas e tem como objetivo apresentar uma solução simples, de baixo custo e bastante eficiente para a resolução de diversos tipos de atividades que possuem essa premissa.

Na execução do projeto foram elaboradas algumas leituras em períodos diferentes do dia e em diferentes dias, com estes dados foram elaboradas algumas tabelas com todas as informações extraídas e seus respectivos gráficos afim de demonstrar as variações existentes ao decorrer do dia. Além da coleta através do ESP32 e sensores, foi feito também alguns cadastros pela interface desenvolvida, bem como algumas consultas para demonstração de sua funcionalidade.

Inicialmente foram coletadas as variáveis de temperatura, luminosidade e umidade durante três dias, onde em cada medição era fornecido ao banco de dados um valor de identificação diferente para poder ser realizadas as consultas a partir do valor da identificação. Este método de busca foi idealizado para um caso onde, por exemplo houvessem a presença de vários dispositivos de coleta de dados trabalhando ao mesmo tempo, facilitando dessa forma uma observação específica dos valores obtidos por cada um e de forma independente.

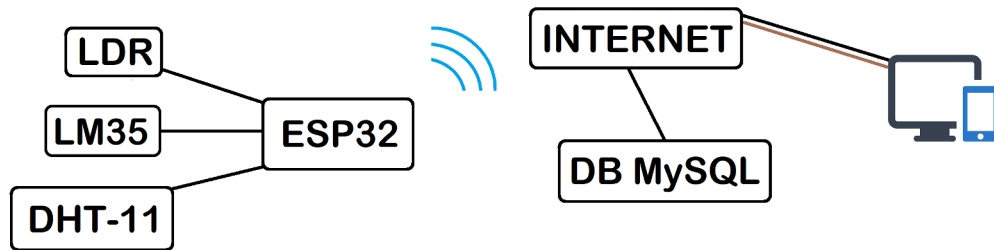
A seguir serão apresentadas as partes que compõem o trabalho, desde a programação da placa de desenvolvimento, o ESP32 aqui utilizado, o desenvolvimento da interface, o sistema de hospedagem até algumas considerações importantes.

3.1 Organização do projeto

Na Figura 8, pode ser visualizada a arquitetura do projeto, basicamente o processo se dá com a captura constante de informações através dos sensores utilizados (LDR, LM35 e DHT-11), a coleta se dá regularmente a partir de um tempo preestabelecido que varia de acordo com a necessidade. Para o projeto em questão o tempo de coleta aplicado é de 20 minutos, dessa forma a cada vinte minutos passados o esp32 fará a leitura e a enviará para o banco MySQL programado para receber as informações. Com os dados já no banco, é possível o acesso, consulta e a adição de informações através de uma interface desenvolvidas em HTML e CSS.

Ao implementar os sensores deve-se atentar a algumas configurações do ESP32, como o fato de 18 das 36 portas lógicas trabalharem com valores analógicos, dessa forma, caso o sensor seja analógico, aplica-lo a uma porta que trabalha apenas com valores digitais o torna inutilizável. Além disso, também há algumas portas analógicas que não funcionam adequadamente quando o microcontrolador é conectado à internet. No projeto foram realizados testes de algumas portas para checar o correto funcionamento, foram então escolhidas as portas de numeração 13, 32 e 35.

Figura 8 – Arquitetura do projeto



Fonte: Autor

3.2 Programação do ESP32

O desenvolvimento e gravação do código foi feito através da IDE do Arduino, para isso foi necessário o *download* e a utilização da biblioteca da placa ESP32 desenvolvida pela própria Espressif Systems. Já para a conexão com a internet e o banco de dados foram utilizadas as bibliotecas WiFi.h, MySQL_Connection.h e MySQL_Cursor.h.

Inicialmente foram incluídas as bibliotecas e definidas as variáveis de login, senha e outras informações úteis para a conexão do ESP com a rede e também com o banco de dados utilizado, nos anexos está presente o código e o trecho específico utilizado. Também foi definida uma variável de comando utilizada na adição de dados ao banco, para isso a variável é definida como um comando em SQL para inserir dados coletados.

A conexão com a internet e o banco de dados também é realizada a partir de comandos específicos, a estrutura de repetição *while* é presente em ambos os trechos e assegura que enquanto não seja conectado, o esp continue tentando até que consiga.

Um detalhe bastante importante presente nos conversores AD de microcontroladores como o ESP32 é a não linearidade de sua leitura para partes da faixa de leitura além de uma diferença de mais ou menos 6% entre chips. Em casos onde a precisão não é um fator chave no desenvolvimento, esse detalhe pode ser passado facilmente, todavia se é necessário um projeto preciso, esse problema poderá ser contornado com a calibração ADC do microcontrolador.

Uma forma de fazer a calibração ADC é com a utilização de um sinal auxiliar, dessa forma ao se fazer uma leitura de um sinal previamente conhecido com o ESP e comparando-os, é possível obter o valor da diferença existente e cada ponto medido. Na Figura 9 é apresentada a medida mencionada onde é possível ver em verde a curva da diferença obtida, que idealmente deveria estar em zero.

Com esta curva é possível obter a função da diferença e com ela elaborar um código onde o próprio ESP realiza a correção de suas leituras.

A parte de *loop* do código realizará a coleta contínua dos dados a cada um terço de hora

Figura 9 – Diferença das curvas AD



Fonte: Fernando K.

e enviará para o banco de dados. Foi elaborado uma condição *if* para o envio desses dados, dessa forma eles serão enviados apenas se a entrada 14 estiver em nível zero. Essa funcionalidade foi implementada para simular uma chave de liga e desliga, assim o ESP continuará coletando dados, mas o envio poderá ser controlado externamente.

Como uma das partes principais do trabalho, a montagem da parte física e também o desenvolvimento do código deve ser bem desenvolvidos afim de evitar erros grosseiros ou simples, mas que possam comprometer o funcionamento do projeto a longo prazo. Mais à frente será relatado alguns problemas ocorridos durante a montagem do circuito e do código.

3.3 Configuração do banco de dados

Para a construção do banco inicialmente foi escolhido o XAMPP, *software* bastante conhecido que junta servidores de códigos abertos como é o caso do MySQL. Foram construídas as tabelas de acordo com os dados coletados e acrescentados alguns campos como o de identificação, o de data e hora e status de dia e noite. A apresentação da tabela ficou como mostrada na Figura 10.

As colunas de Humidade, Luminosidade e Temperatura são obtidas a partir dos sensores, a de Status é definida a partir do algoritmo presente no ESP32 que utiliza a variável Luminosidade para sua definição.

Com as informações no banco foi possível fazer ações de buscas inteligentes, separar as informações por períodos, plotagens de gráficos de vários tipos diferentes, transforma-los em figuras entre outras coisas. Para permitir o acesso externo do banco foi definido um usuário e uma senha com algumas permissões básicas de adição, exclusão e consultas de dados.

Figura 10 – Configurações do banco de dados

The screenshot shows a database management tool interface. At the top, there's a navigation bar with buttons: Procurar, Estrutura, SQL, Pesquisar, Inserir, Exportar, Importar, Privilegios, Operações, and Ras. Below this, there are tabs for 'Estrutura da tabela' and 'Visão de relação(ões)'. The main area displays a table structure for 'Tabela: dados'.

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Ações
1	ID	int(11)			Não	Nenhum			Muda, Elimina, Mais
2	Luminosidade	float			Não	Nenhum			Muda, Elimina, Mais
3	Temperatura	float			Não	Nenhum			Muda, Elimina, Mais
4	Umidade	float			Não	Nenhum			Muda, Elimina, Mais
5	Status	varchar(10)	utf8mb4_general_ci		Não	Nenhum			Muda, Elimina, Mais
6	Data	datetime			Não	current_timestamp()			Muda, Elimina, Mais

Below the table, there are options to 'Marcar todos', 'Com os seleccionados', and buttons for 'Procurar', 'Muda', 'Elimina', 'Primária', 'Único', 'Índice', and 'Espacial'. At the bottom, there are buttons for 'Texto Completo', 'Adicionar à(s) coluna(s) central(is)', and 'Remover da(s) coluna(s) central(is)'.

Fonte: Autor.

3.4 Desenvolvimento da interface

Para a o desenvolvimento do ambiente *front-end* foram utilizadas as linguagens HTML e CSS para criação de seu *design*. No desenvolvimento *back-end* foi utilizado códigos PHP para ligação entre página e banco. A Figura 11 mostra a página modelo desenvolvida e utilizada. Assim como o código em C utilizado na programação do ESP32, o que foi utilizado na interface está disposto nos anexos, tanto o HTML e CSS quanto o PHP.

Figura 11 – Interface desenvolvida

The screenshot shows a web interface for database access. At the top, there is a logo for 'UNIVERSIDADE FEDERAL UFERSA RURAL DO SEMI-ÁRIDO'. Below the logo, the text 'Acesso ao banco de dados' is displayed. There are two main buttons: 'Fazer consulta' and 'Ir ao banco'. Below these buttons, the text 'Inserir abaixo dados ao banco' is shown. There are four input fields for data entry: 'Identificação', 'Luminosidade', 'Temperatura', and 'Umidade'. At the bottom, there is a button labeled 'Inserir dados'.

Fonte: Autor

3.5 Configurações para o acesso externo

Para realizar o acesso externo ao sistema de monitoramento criado, é necessário fazer algumas configurações. No trabalho foram testados dois tipos de acesso, um através da configura-

ção de um endereço de IP externo onde era possível o acesso à máquina através desse endereço. A segunda forma realizada foi através da contratação de um serviço de host para manter a página disponível 24 horas. A segunda opção se mostrou mais viável do que a primeira, tendo em vista que o endereçamento com o IP só seria possível se a máquina estivesse ligada direto. Já com um serviço de host é possível acessar à página a qualquer momento sem essa necessidade, já que a página estaria em outro servidor.

Para realizar a hospedagem de um site, existem hoje diversas opções de serviços no mercado, para o trabalho foi utilizado a plataforma Umbler. Entretanto, todas funcionam basicamente da mesma forma. Além dos arquivos que devem ser dispostos, há também a necessidade de um domínio para que esses arquivos que geram a página possam ser acessados. Um domínio nada mais é do que um nome utilizado para identificar e localizar conjuntos de computadores na internet, dessa forma é possível a memorização desses caminhos de forma fácil sem que seja necessário a utilização de grandes sequencias de números. Alguns exemplos de domínios são: .com.br, .com e .net.

Os domínios adquiridos podem ser pagos ou não. Existem alguns domínios menos usuais como .ga e .tk que são oferecidos por alguns serviços. No projeto foi selecionado o domínio iottcc.tk e hospedada a página. Quando acessado o endereço, a tela de login apresentava o sistema mostrado na Figura 12 abaixo.

Figura 12 – Projeto hospedado



Fonte: Autor

4 RESULTADOS E DISCUSSÕES

Apesar da forma de tratamento de dados escolhida e implementada no trabalho, uma série de tipos de buscas podem ser implementadas com o auxílio do dispositivo ESP32, como uma seleção de valores específicos tais como temperaturas limites, horários (através das leituras do LDR ou da data e hora do banco) e alguns *status* elaborados pelo próprio desenvolvedor.

4.1 Montagem e programação do ESP32

No desenvolvimento do trabalho não foi necessário realizar a correção precisa do conversor AD já que a precisão não foi algo tão visado, contudo na utilização do sensor LM35 foi observado uma diferença nos valores medidos, para isso foi feita uma pequena correção baseada no procedimento apresentado com as medidas realizadas a partir do auxílio de um multímetro. O trecho de código na Figura 13 mostra a aplicação de um fator corretivo no valor de 1,9867.

Figura 13 – Correção AD no LM35

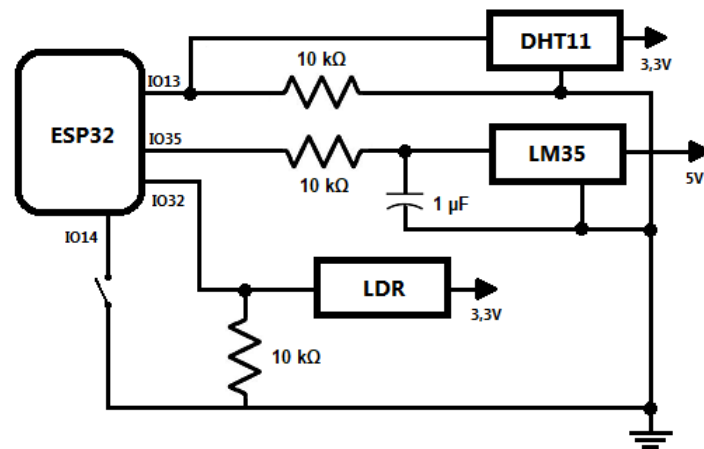
```
void loop() {
    delay(1000);
    Serial.println(analogRead(35)*1.9867);
    delay(100);
    Serial.println(analogRead(35)*(1.9867*3.3/4095)/0.01);
    if(int (digitalRead(14)) == 0)
```

Fonte: Autor

O circuito foi composto por um dispositivo ESP32, um sensor de temperatura e umidade DHT11, um sensor LDR, um sensor de temperatura LM35, uma chave ON/OFF, resistores 10 k Ω , capacitor 1 μ F, uma protoboard e alguns fios de ligação. Os sensores foram ligados nas portas 13, 14, 32 e 35 e alimentados pelos pinos de 3,3V e 5V presentes no próprio microcontrolador. A chave ON/OFF foi utilizada para o controle do fluxo de dados no MySQL, dessa forma, caso a chave fosse ligada os dados eram enviados, caso desligada haveria apenas a leitura sem o envio. A presença do capacitor é necessária para atuar como um filtro e assim evitar erros na leitura. Na Figura 14 é possível ver os esquema de ligação utilizado, já a 15 é possível ver a foto do sistema real utilizado.

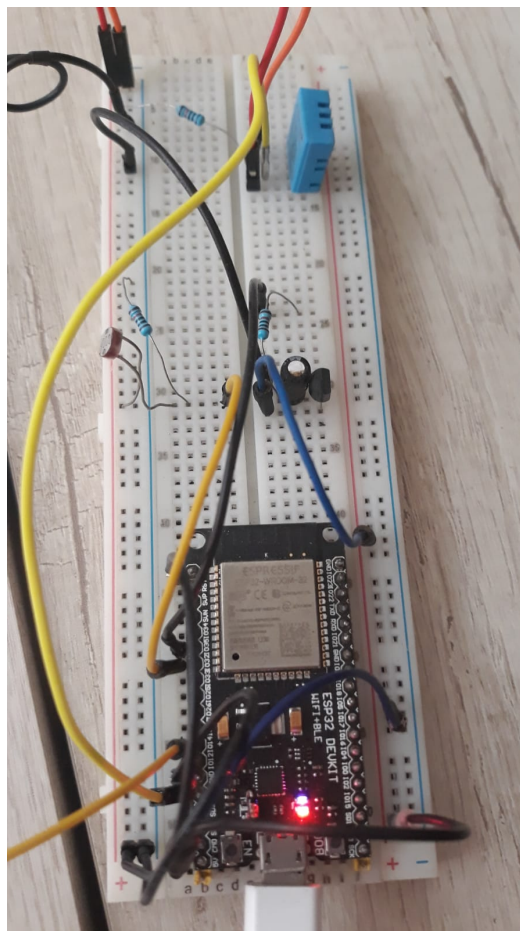
Depois de montado o circuito, foram feitos testes para obtenção de parâmetros utilizados na programação. O primeiro teste foi feito com o LDR, foram elaboradas leituras durante o dia para que se pudesse achar valores que correspondessem ao amanhecer e anoitecer. A partir de uma estimativa baseada nos testes, foram definidos estes valores. Também foram realizados testes para o LM35 e neles foi observado uma discrepância entre o seu valor emitido e o que era lido pelo ESP32, isso se deu pela não precisão dos conversores ADC como já mencionado no trabalho. Essa constatação foi feita através do *Plotter Serial* da IDE do Arduino em comparação com valores apresentados por um multímetro. No DHT11 não foi necessário a utilização correções,

Figura 14 – Circuito do projeto



Fonte: Autor

Figura 15 – Circuito real do projeto

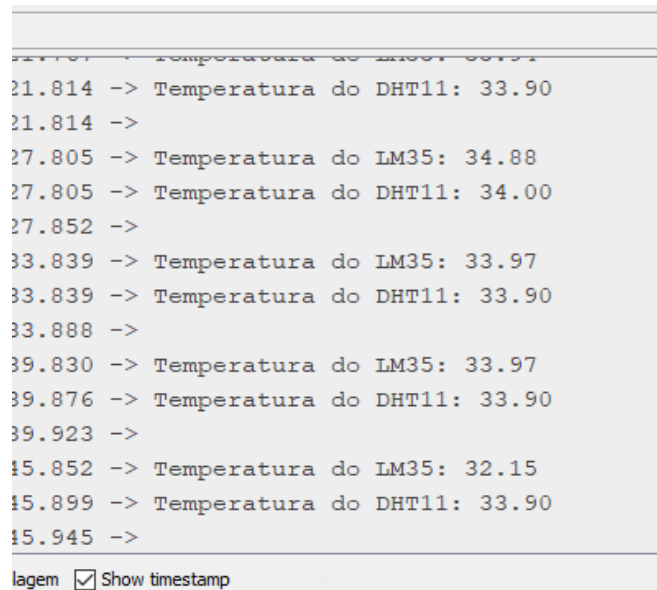


Fonte: Autor

mas como também trabalha com a leitura de temperatura, foi utilizado para comparação com o LM35 após sua correção.

Como pode ser visto, a correção feita na programação apresentou resultados bem próxi-

Figura 16 – Comparação entre valores do DHT11 e LM35.



Fonte: Autor

mos aos enviados pelo DHT11.

O código utilizado para programação do ESP32 foi desenvolvido na IDE do Arduino, que por sua vez faz uso da linguagem C. Será apresentado alguns trechos principais como os de conexão com a rede e o MySQL, de definições e de decisões.

Inicialmente, foram definidas as bibliotecas utilizadas. Duas delas são utilizadas para a conexão e comunicação com o banco de dados e as outras duas para a conexão com a WiFi e utilização dos sensores DHT. Posteriormente houve as definições de login e senha da WiFi e do banco e o IP.

Figura 17 – Bibliotecas e definições utilizada.

```
//INCLUSÃO DE BIBLIOTECAS UTILIZADAS
#include <MySQL_Connection.h>
#include <MySQL_Cursor.h>
#include <WiFi.h>
#include "DHT.h"

//DEFINIÇÕES UTILIZADAS
#define wifi_login "*****"
#define wifi_senha "*****"
#define server_login "*****"
#define server_senha "*****"

//VARIÁVEIS PARA MySQL
IPAddress server_addr (000, 000, 000, 000);
char user[] = server_login;
char password[] = server_senha;
```

Fonte: Autor

Na Figura 18 é visto a parte posterior do código que trata de uma definição de comando em PHP que será enviado para o banco, este comando seleciona as colunas e insere nelas valores pré definidos, nesse caso foi posto o ID como valor inteiro, a Luminosidade, Temperatura e Umidade foram definidas como flutuante e a de Status como caractere. Também é observado a definição do tipo cliente e a instância de objeto.

Figura 18 – Comandos PHP e instância de objeto

```
//COMANDO UTILIZADO PARA A ADIÇÃO DOS DADOS
char INSERT_SQL[] = "INSERT INTO iot_project.dados "
"(ID, Luminosidade, Temperatura, Umidade, Status) VALUES ('%d', '%f', '%f', '%f', '%s')";
char query[150];

//INSTÂNCIA DE OBJETOS
WiFiClient client;
MySQL_Connection conn((Client *)&client);
```

Fonte: Autor

Os próximos blocos desenvolvidos foram os que efetivamente realizam as conexões entre o ESP32 e o banco de dados e WiFi, neles há a presença do comando *while* que cria uma laço onde a saída do mesmo só acontecerá com a conexão bem sucedida, enquanto não há a conexão, será mostrado um texto com o informe e um intervalo de 500 ms entre novas tentativas. A função "Serial.println" é utilizado para uma melhor visualização do andamento do processo e apresenta algum aviso geralmente através do Serial Plotter ou de *displays*.

Passando para a parte do *loop*, foi definido o tempo de 20 minutos (1.200.000 ms) entre cada coleta de dados, onde com o auxílio da função "millis();" foi feito esse controle de tempo. A vantagem de usar essa função é que o microprocessador não para de executar a contagem de tempo, diferente da função "delay();". A cada período de tempo os sensores eram lidos e armazenados em variáveis. Também foi elaborado uma condição *if* para utilização de uma chave ON/OFF, dessa forma foi possível fazer a ativação e desativação do envio dos dados a qualquer momento. Quando ativado, os dados armazenados nas variáveis passavam por outros comandos responsáveis por organiza-los e envia-los ao comando em PHP definido na Figura 18.

Figura 19 – Conexões com o WiFi e DB

```

//CONECTA COM A INTERNET
Serial.println();
Serial.println();
Serial.print("Connecting to ");
Serial.println(ssid);

WiFi.begin(ssid, passwordwifi);

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}

Serial.println("");
Serial.println("WiFi connected.");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

//CONECTAR MySQL
while(!conn.connect(server_addr, 3306, user, password))
{
    Serial.println("Conexão SQL falhou.");
    conn.close();
    delay(1000);
    Serial.println("Conectando SQL novamente.");
}

Serial.println("Conectado ao Servidor SQL.");

```

Fonte: Autor

Figura 20 – Coleta e envio dos dados

```

void loop()
{
    if ((millis()-tempo)>1200000)
    {
        a=analogRead(32); // REALIZA A LEITURA DO LDR
        b = analogRead(35)*(1.882*3.3/4095)/0.01; // REALIZA A LEITURA DO LM35
        h = dht.readHumidity(); // REALIZA A LEITURA DO DHT11

        if(a<400)
        {
            Status = "Noite";
        }
        if(a>1000)
        {
            Status = "Dia";
        }

        if(digitalRead(14) == 0) // LIGA/DESLIGA ENVIO DE DADOS
        {
            sprintf(query, INSERT_SQL, 5, a, b, h, Status);
            MySQL_Cursor *cur_mem = new MySQL_Cursor(&conn);
            cur_mem->execute(query);
            delete cur_mem;
            Serial.println("Informações enviadas");
            Serial.println(" ");
        }
        tempo = millis();
    }
}

```

Fonte: Autor

4.2 Coleta dos dados

No total, foram realizadas 5 medições de períodos do dia, essas medições foram feitas em dias e períodos diferentes como já mencionado. A primeira medição foi realizada as 08:10 até as 17:16 do dia 02 de agosto. No mesmo dia foram coletados dados do período compreendido entre 19:51 até as 06:11 da manhã do dia seguinte. Estes valores obtidos estão dispostos na Figura 21 abaixo.

Figura 21 – Dados coletados ID: 1 e 2

ID	Luminosidade	Temperatura	Umidade	Status	Data	ID	Luminosidade	Temperatura	Umidade	Status	Data
1	1885.2	19.4129	55	Dia	2021-08-02 08:10:41	2	0	28.5126	42	Noite	2021-08-02 19:51:07
1	2024.4	19.5645	55	Dia	2021-08-02 08:30:41	2	0	28.816	42	Noite	2021-08-02 20:11:07
1	2127.4	20.3228	55	Dia	2021-08-02 08:50:41	2	0	28.6643	43	Noite	2021-08-02 20:31:08
1	2235.2	20.9295	53	Dia	2021-08-02 09:10:42	2	0	28.816	43	Noite	2021-08-02 20:51:08
1	2346.4	20.4745	53	Dia	2021-08-02 09:30:42	2	0	27.2993	46	Noite	2021-08-02 21:11:08
1	2446	20.3228	52	Dia	2021-08-02 09:50:42	2	0	27.6027	45	Noite	2021-08-02 21:31:08
1	2468.2	20.7778	51	Dia	2021-08-02 09:56:19	2	0	27.7543	46	Noite	2021-08-02 21:51:09
1	2542.4	20.3228	51	Dia	2021-08-02 10:16:19	2	0	27.6027	47	Noite	2021-08-02 22:11:09
1	2576.8	21.6878	50	Dia	2021-08-02 10:36:19	2	0	27.1477	48	Noite	2021-08-02 22:31:10
1	2510.8	21.8395	49	Dia	2021-08-02 10:56:20	2	0	26.6927	50	Noite	2021-08-02 22:51:10
1	2627	21.8395	48	Dia	2021-08-02 11:16:21	2	0	26.8444	52	Noite	2021-08-02 23:11:11
1	2624.6	22.4461	47	Dia	2021-08-02 11:36:21	2	0	26.541	51	Noite	2021-08-02 23:31:11
1	2603.4	23.2044	47	Dia	2021-08-02 11:56:21	2	0	26.6927	52	Noite	2021-08-02 23:51:12
1	2541.6	23.5078	46	Dia	2021-08-02 12:16:21	2	0	26.6927	53	Noite	2021-08-03 00:11:12
1	2461	24.1144	45	Dia	2021-08-02 12:36:22	2	0	26.2377	53	Noite	2021-08-03 00:31:12
1	2401.4	24.7211	45	Dia	2021-08-02 12:56:22	2	0	26.3894	54	Noite	2021-08-03 00:51:13
1	2351	24.8727	44	Dia	2021-08-02 13:16:23	2	0	25.7827	55	Noite	2021-08-03 01:11:13
1	2319.8	25.3277	42	Dia	2021-08-02 13:36:23	2	0	25.631	55	Noite	2021-08-03 01:31:14
1	2285	26.2377	42	Dia	2021-08-02 13:56:23	2	0	25.0244	55	Noite	2021-08-03 01:51:14
1	2294.6	26.3894	40	Dia	2021-08-02 14:16:23	2	0	24.5694	57	Noite	2021-08-03 02:11:14
1	2328.8	26.996	40	Dia	2021-08-02 14:36:24	2	0	25.0244	57	Noite	2021-08-03 02:31:15
1	2313	27.1477	39	Dia	2021-08-02 14:56:24	2	0	24.4177	57	Noite	2021-08-03 02:51:16
1	2462.8	27.2993	40	Dia	2021-08-02 15:16:24	2	0	24.5694	58	Noite	2021-08-03 03:11:16
1	2571.8	28.361	38	Dia	2021-08-02 15:36:25	2	0	23.8111	58	Noite	2021-08-03 03:31:17
1	842.8	28.361	39	Dia	2021-08-02 15:56:26	2	0	24.1144	58	Noite	2021-08-03 03:51:17

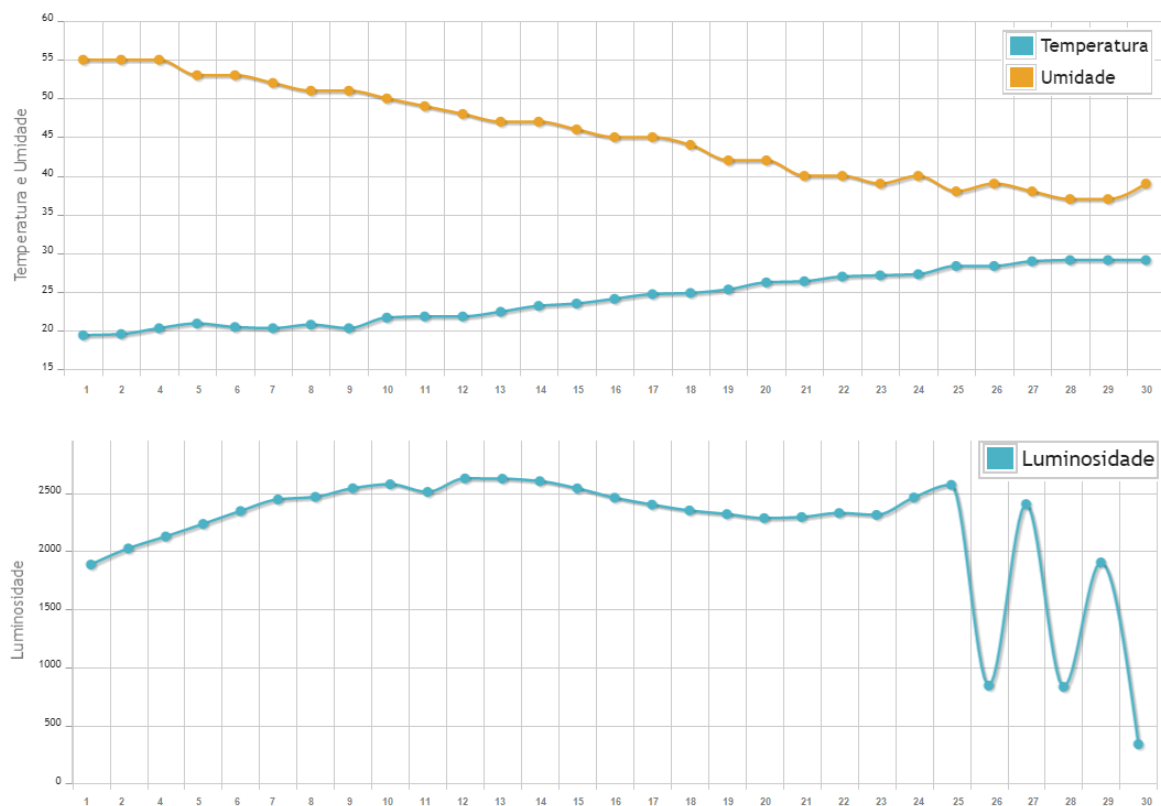
Fonte: Autor

Ao trabalhar com bancos MySQL, é possível fazer várias coisas com os dados nele contido, uma funcionalidade bastante útil do MySQL é a possibilidade de plotar gráficos referentes aos seus dados. Afim de facilitar a visualização das informações, essa funcionalidade será utilizada no lugar das tabelas. Os valores obtidos na medição com ID 1 da tabela da Figura 21 estão demonstrados em forma de gráfico na Figura 22.

Como pode ser observado nos gráficos, os valores obtidos mostram coerência com a evolução temporal onde valores de temperatura apresentaram valores crescentes e a umidade por sua vez apresentou valores decrescente. Já que o ambiente possui essas características de maiores temperaturas e menor umidade ao entardecer, não foi constatado anormalidades nos valores medidos. No gráfico referente a luminosidade é possível ver poucas oscilações durante o período de coleta, exceto pelo trecho final do gráfico onde é possível ver uma oscilação, esta é gerada devido o conversor AD do ESP32 ser bastante sensível, contando com 4095 posições diferentes em seu range (12 bits), dessa forma, entre cada posição há uma variação aproximada de 0,8 mV. É valido destacar que a medição da iluminação foi feita a partir dessa variação de 12

bits.

Figura 22 – Gráfico da Temperatura, Umidade e Luminosidade (ID 1)



Fonte: Autor

Dessa forma, qualquer variação existente na luminosidade do ambiente como por exemplo o acionamento de uma luz a noite ou o bloqueio dos raios solares por nuvens irá ser facilmente detectada.

As medidas de identificação 4 e 5 foram obtidas em horários de transição entre dia e noite, dessa forma foi possível observar uma outra coluna desenvolvida na criação da tabela, essa outra coluna é a *Status*, onde nela foram postos os valores de "Dia" e "Noite". Esta função foi implementada a partir do próprio dispositivo ESP32 com decisões sobre a luminosidade. Como já mencionado, muitos processos de análises e tomadas de decisões podem ser feitas com o ESP32 e isso implica na possibilidade de utilização de dados registrados em tomadas de decisões sem que não seja necessário que estes dados estejam gravados dentro do dispositivo. Também é possível utilizar estes dados como uma parâmetro de busca, dessa forma é possível fazer uma análise sobre as variações ambientais apenas durante o período do dia ou da noite.

Como é visto na Figura 23, a transição de dia para noite aconteceu na leitura feita por volta das 17:06, este horário de fato não pertence ao período da noite (18:00 às 23:59), contudo é possível realizar uma correção do valor de acordo com o ambiente que os sensores estão presentes. No código do ESP32 foi elaborada uma lógica onde valores abaixo de 400 definiria o estado como noite e acima de 1000, o estado passaria a ser dia. Esses valores foram definidos a

Figura 23 – Gráfico da Temperatura, Umidade e Luminosidade (ID 4)

ID	Luminosidade	Temperatura	Umidade	Status	Data
4	1280.2	35.0342	20	Dia	2021-08-07 16:06:20
4	1053.6	30.9393	25	Dia	2021-08-07 16:26:20
4	621.2	29.8776	62	Dia	2021-08-07 16:46:26
4	176.8	29.4226	59	Noite	2021-08-07 17:06:49
4	590.4	29.271	39	Noite	2021-08-07 17:26:49
4	561.4	29.1193	39	Noite	2021-08-07 17:47:14

Fonte: Autor

partir de alguns testes, mas ao realizar a coleta em um local um pouco distante de onde foram feitos esses testes, erros foram gerados.

Na Figura 24 é observada a tabela com os valores de transição de noite para dia, nesta a transição ocorre perto das 07:00 horas. Seu primeiro valor de luminosidade é nulo devido a inexistência de luz. O que não ocorre nos valores da Figura 23 devido a presença de lâmpadas acesas, dessa forma é possível ver um aumento deixando os dois últimos valores na faixa dos 500.

Figura 24 – Tabela da Temperatura, Umidade e Luminosidade (ID 5)

ID	Luminosidade	Temperatura	Umidade	Status	Data
5	0	26.8444	61	Noite	2021-08-09 05:39:20
5	135.4	27.7543	59	Noite	2021-08-09 05:59:20
5	446.2	26.6927	62	Noite	2021-08-09 06:19:24
5	752.2	27.6027	62	Noite	2021-08-09 06:39:24
5	1155.4	27.6027	59	Dia	2021-08-09 06:59:25
5	872.6	26.996	56	Dia	2021-08-09 07:19:25

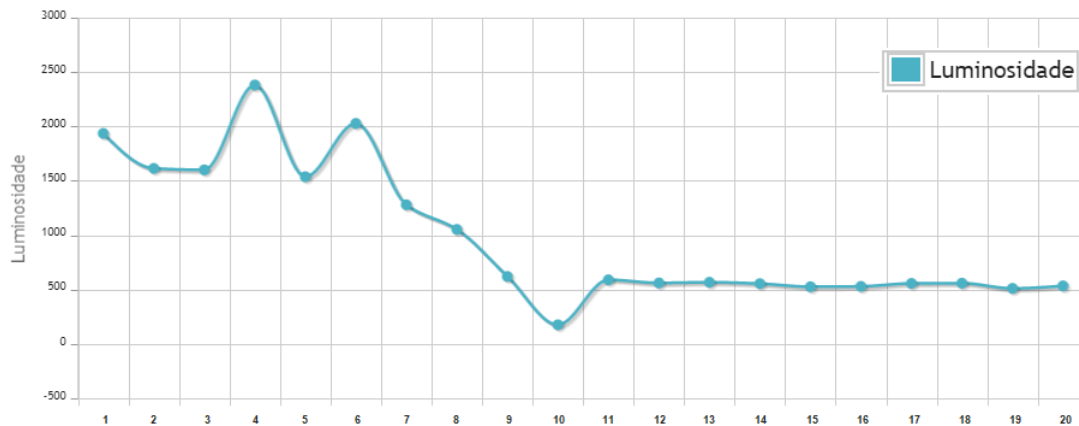
Fonte: Autor

4.3 CADASTRO E CONSULTA PELA PÁGINA CRIADA

No processo de criação da página onde há a interação do usuário com o banco, foi idealizado um ambiente simples e intuitivo que houvesse a possibilidade de redirecionamento ao banco, a inclusão e a consulta a dados existentes. Há dois termos utilizados na criação desse tipo de páginas, o *front-end* e o *back-end*, o primeiro refere-se a parte visual e interativa de um site, o segundo está ligado a parte que faz a conexão com o banco de dados.

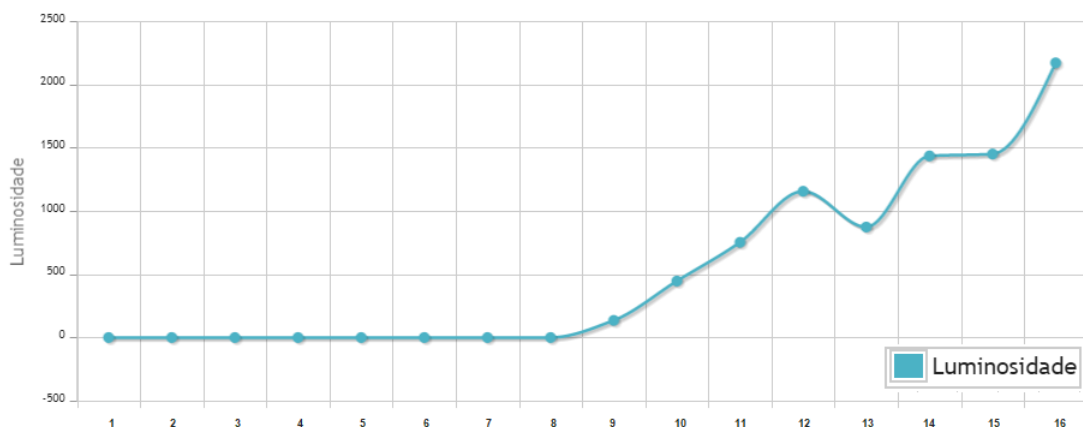
Também foi pensado em um sistemas de cadastros, para isso foi feito um sistema capaz de cadastrar as próprias variáveis coletadas pelos sensores. Foram desenvolvidos alguns códigos para a montagem da página, para o seu estilo (cores, tamanhos, formatos, espaçamentos...) e

Figura 25 – Tabela da Luminosidade (ID 4)



Fonte: Autor

Figura 26 – Gráfico da Luminosidade (ID 5)



Fonte: Autor

também para a conexão com o banco de dados. Estes códigos estão dispostos nos anexos e não serão explorados aqui por conveniência.

Os testes realizados se mostraram eficientes e corresponderam com as expectativas. Inicialmente foi feito o cadastro de um dado qualquer e observado se o mesmo seria enviado corretamente ao MySQL ou se apresentaria algum tipo de erro. Foi escolhido uma identificação de valor 7, uma luminosidade de valor 1000, 28°C para a temperatura e 40% de humidade.

Pela confirmação de envio, os dados foram enviados com sucesso ao banco, o que foi confirmado com a consulta feita logo em seguida. Como os testes foram feitos depois da coleta, também foi possível ver os valores obtidos através do ESP32 32 durante sua atuação. As Figuras 29 e 30 mostram respectivamente as consultas pelos valores 7 e 5 de identificação.

Como é possível ver, o valor obtido na Figura 29 foi o cadastrado pela página, já na 30 os valores são os coletados pelo ESP32 e podem ser comparados através do gráfico presente na

Figura 27 – Enviando dados ao banco pela página HTML



UNIVERSIDADE FEDERAL
UFERSA
RURAL DO SEMI-ÁRIDO

Acesso ao banco de dados

Fazer consulta

Ir ao banco

Inserir abaixo dados ao banco

7	1000
28	40

Inserir dados

Fonte: Autor

Figura 28 – Enviando dados ao banco pela página HTML



UNIVERSIDADE FEDERAL
UFERSA
RURAL DO SEMI-ÁRIDO

Dados inseridos com sucesso!

Voltar

Fonte: Autor

Figura 26. Também é possível observar a ausência do Status no dado cadastrado, isso ocorreu pois o mesmo é definido pelo próprio microcontrolador em uma de suas instruções, mas poderia ser implementado um cadastro para a coluna de Status se fosse desejado.

Como mencionado, foi idealizado um sistema de buscas que realizasse as ações a partir da identificação, contudo seria possível ampliar o sistema de buscas e padroniza-lo de formas diferentes e que viesse a atender as funções desejadas.

A terceira opção fornecida pela página era de ir direto ao banco de dados, onde nele seria possível outras ações como a geração de gráficos, a limpeza de dados, a modificação da estrutura atual entre outras coisas.

Figura 29 – Consulta ID 7



UNIVERSIDADE FEDERAL
UFERSA
RURAL DO SEMI-ÁRIDO

Consultas

Filtrar:

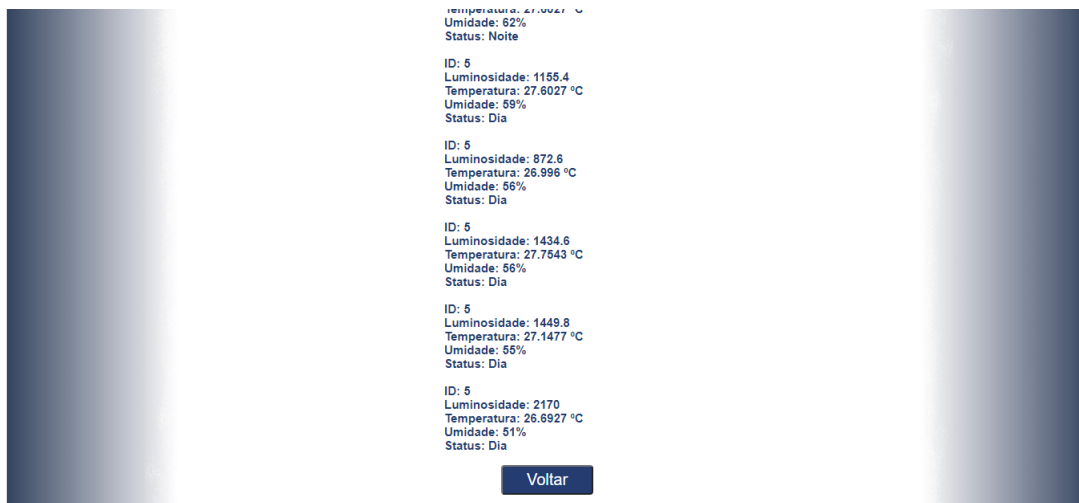
Pesquisar

ID: 7
Luminosidade: 1000
Temperatura: 28 °C
Umidade: 40%
Status:

Voltar

Fonte: Autor

Figura 30 – Valores finais (ID 5)



Temperatura: 27.6027 °C
Umidade: 62%
Status: Noite

ID: 5
Luminosidade: 1155.4
Temperatura: 27.6027 °C
Umidade: 59%
Status: Dia

ID: 5
Luminosidade: 872.6
Temperatura: 26.996 °C
Umidade: 56%
Status: Dia

ID: 5
Luminosidade: 1434.6
Temperatura: 27.7543 °C
Umidade: 56%
Status: Dia

ID: 5
Luminosidade: 1449.8
Temperatura: 27.1477 °C
Umidade: 56%
Status: Dia

ID: 5
Luminosidade: 2170
Temperatura: 26.6927 °C
Umidade: 51%
Status: Dia

Voltar

Fonte: Autor

Trabalhos Futuros

Varias propostas surgem ao se pensar em trabalhos futuros devido a vasta possibilidade que a internet das coisas oferece, mas para o trabalho em questão é possível melhorar e também adicionar algumas coisas. A respeito do sistema de coleta de dados, é interessante destina-lo a alguma função específica, as opções são várias, como exemplo pode ser citados a monitoração de uma estufa ou de uma chocadeira, também pode ser utilizados sensores diferentes e utilizado para a verificação de nível, monitoramento de humidade do solo em projetos de irrigação entre outras possibilidades. Além do direcionamento da monitoração, é possível implementar um

sistema de controle que possa estabilizar as variáveis em um ponto pré determinado, que por sua vez também pode ser escolhido através do sistema de cadastro implementado.

A respeito do site criado, também podem ser implementadas funções diferentes afim de satisfazer as necessidades do projeto. Existem diversas aplicações possíveis de realizar que podem tornar a usabilidade algo fácil e interativo, um exemplo disso é a responsabilidade que ao ser implementada nos códigos acaba gerando uma facilidade de uso quando acessado o site através de um celular, por exemplo. Também podem ser utilizados códigos que no lugar de apresentar os dados um a um como foi feito, gerar gráficos com os valores do banco de dados, esse tipo de apresentação pode ser implementado com a utilização de JavaScript que é uma linguagem de programação estruturada de scripts de alto nível. Por fim e não menos importante, a segurança da página também é algo essencial e que deve ser prezada, dessa forma, caso seja um projeto bastante sensível no quesito segurança e que possa vir a trazer grandes danos e perdas, a implementação de códigos que visem a segurança é algo que deve ser considerado.

5 CONCLUSÃO

Com a finalização do trabalho, foi possível alcançar todos os objetivos propostos, que consistiam em desenvolver e implementar com sucesso uma aplicação da *Internet of Things* de uma forma simples, acessível e eficiente, onde realizasse tanto a coleta quanto o armazenamento de dados de um ambiente, afim de demonstrar a viabilidade e a facilidade de implementação da tecnologia em processos e atividades, fazer uma abordagem do tema apresentando a sua importância e perspectivas futuras, com cada vez mais sua presença aumentando e os dispositivos e objetos se tornando mais interligados uns aos outros e por fim, realizar uma integração entre o sistema de coleta elaborado com uma interface gráfica que pudesse fazer uma comunicação homem-máquina, de forma que fosse possível a visualização de dados e também a inserção. Espera-se que o presente trabalho contemple os conhecimentos básicos da IoT e seja um meio de incentivar os leitores a se interessarem por este tema tão abrangente e de tanta perspectiva de crescimento.

REFERÊNCIAS

Adrobotica. Fotorresistor - Sensor De Luz LDR 5mm. Disponível em <<https://www.adrobotica.com/produto/fotorresistor-sensor-de-luz-ldr-5mm/>>. Acessado em 01 de Nov. 2021.

ASAIR. DHT11. 2017. Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/1132088/ETC2/DHT11.html>>. Acesso em: 21 nov. 2021.

ASSOCIAÇÃO BRASILEIRA DE INTERNET DAS COISAS. **Previsões para o Mercado de IOT**. 2021. Disponível em: <https://abinc.org.br/previsoes-para-omercado-de-iot/>. Acesso em: 27 Ago. 2021.

FILIPEFLOP. Sensor de Temperatura LM35DZ. Disponível em <<https://www.filipeflop.com/produto/sensor-de-temperatura-lm35dz/>>. Acessado em 01 de Nov. 2021.

GALEGALE, Gustavo Perri et al. INTERNET DAS COISAS APLICADA A NEGÓCIOS: UM ESTUDO BIBLIOMÉTRICO. **Revista de Gestão da Tecnologia e Sistemas de Informação**, Universidade de São Paulo, v. 13. 15 p, 03 Dez. 2016.

KOYANAGI, Fernando. **Você não sabia? Ajuste de ADC do ESP32**. fernandok. Internet, 2018. Disponível em: <https://www.fernandok.com/>. Acesso em: 27 ago. 2021.

LIMA, Adilson da Silva. **MySQL server: versões open source 4.X : soluções para desenvolvedores e administradores de bancos de dados**, f. 199. 2002. 398 p.

MYSQL. **MySQL 8.0 Reference Manual**. MySQL. Disponível em: <https://dev.mysql.com/doc/refman/8.0/en/>. Acesso em: 27 ago. 2021.

ORACLE. **O Que É um Banco de Dados?**. Oracle. Disponível em: <https://www.oracle.com/br/database/what-is-database/>. Acesso em: 30 ago. 2021.

PATEL, Keyur K.; PATEL, Sunil M. **Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application Future Challenges**. 2016. 10 p. Disponível em: https://www.researchgate.net/publication/330425585_Internet_of_Things-IOT_Definition_Characteristics_Architecture_Enabling_Technologies_Application_Future_Challenges. Acesso em: 27 ago. 2021.

SANTOS, JEAN WILLIAN; JUNIOR, RENATO CAPELIN DE LARA. **SISTEMA DE AUTOMATIZAÇÃO RESIDENCIAL DE BAIXO CUSTO CONTROLADO PELO MICRO-CONTROLADOR ESP32 E MONITORADO VIA SMARTPHONE**. PONTA GROSSA, 2019 Trabalho de Conclusão de Curso (Tecnologia em Automação Industrial) - Universidade Tecnológica Federal do Paraná.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S.. **Sistema de banco de dados**, f. 391. 2005. 781 p.

SOUSA, DIONE CARLOS FERREIRA DE et al. **SISTEMA SUPERVISÓRIO COMO AUXILIADOR INTERATIVO DO PROCESSOS DE TIRAGEM DO CAFÉ EXPRESSO** Trabalho de Conclusão de Curso (Tecnologia em Automação Industrial) - Faculdade de Tecnologia de Tatuí - Prof. Wilson Roberto Ribeiro de Camargo.

Espressif Systems. **ESP32 Datasheet**. 2016. Disponível em: <https://cdn.sparkfun.com/datasheets/IoT/esp32_datasheet_en.pdf>. Acesso em: 21 nov. 2021.

Espressif Systems. **ESP32 Series**. 2021. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Acesso em: 21 nov. 2021.

THOMAZINI, Daniel; ALBUQUERQUE, Pedro U. B. **Sensores industriais: Fundamentos e aplicações**. Ed.: érica, 4ª ed. 2009

VERMESAN, Dr. Ovidiu; FRIESS, Dr. Peter. **Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems**. River Publishers, v. 3, f. 182, 2013. 364 p.

6 APÊNDICES

APÊNDICE A — Código utilizado para programar o ESP32

```
//INCLUSÃO DE BIBLIOTECAS UTILIZADAS

#include <MySQL_Connection.h>

#include <MySQL_Cursor.h>

#include <WiFi.h>

#include "DHT.h"

//DEFINIÇÕES UTILIZADAS

#define wifi_login "*****"

#define wifi_senha "*****"

#define server_login "*****"

#define server_senha "*****"

//VARIÁVEIS PARA MySQL

IPAddress server_addr (000, 000, 000, 000);

char user[] = server_login;

char password[] = server_senha;

//COMANDO UTILIZADO PARA A ADIÇÃO DOS DADOS

char INSERT_SQL[] = "INSERT INTO iot_project.dados "

"(ID, Luminosidade, Temperatura, Umidade, Status) VALUES ('%d',

'%f', '%f',

'%f', '%s')";

char query[150];

//INSTÂNCIA DE OBJETOS

WiFiClient client;

MySQL_Connection conn((Client*)&client);

float a, b, a1, h;

int cont=0;

String Status;

unsigned long tempo = 1200000;
```

```
#define DHTTYPE DHT11 // DHT 11

DHT dht(13, DHTTYPE);

void setup() {
  Serial.begin(9600);

  //CONECTA COM A INTERNET

  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(wifi_login);
  WiFi.begin(wifi_login, wifi_senha);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  } Serial.println();
  Serial.println("WiFi connected.");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());

  //CONECTAR MySQL

  while(!conn.connect(server_addr, 3306, user, password))
  { Serial.println("Conexão SQL falhou.");
    conn.close();
    delay(1000);
    Serial.println("Conectando SQL novamente."); }
  Serial.println("Conectado ao Servidor SQL.");

  pinMode(35, INPUT);
  pinMode(14, INPUT_PULLUP);
  pinMode(32, INPUT);
  dht.begin(); }

void loop() {
  if ((millis()-tempo)>1200000) {
```

```
for(cont=0; cont<5; cont++) // LUMINOSIDADE MÉDIA AMBIENTE (LEITURA)
{
    a1 = analogRead(32);
    a += a1; }
a=a/5;
b = analogRead(35)*(1.882*3.3/4095)/0.01; // REALIZA A LEITURA DO LM35
h = dht.readHumidity(); // REALIZA A LEITURA DO DHT11
if (isnan(h)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return; }
Serial.println(a); // IMPRIME OS VALORES NA TELA
Serial.println(b);
Serial.println(h);
Serial.println();
if(a<400) {
    Status = "Noite"; }
if(a>1000) {
    Status = "Dia"; }
if(digitalRead(14) == 0) // LIGA/DESLIGA ENVIO DE DADOS
{
    sprintf(query, INSERT_SQL, 5, a, b, h, Status);
    MySQL_Cursor *cur_mem = new MySQL_Cursor(&conn);
    cur_mem->execute(query);
    delete cur_mem;
    Serial.println("Informações enviadas");
    Serial.println();
}
tempo = millis(); }
a=0;

//RECONECTA AO DB CASO CAIA
while(!conn.connect(server_addr, 3306, user, password))
{
    Serial.println("Reconectando DB");
```

```
conn.close();  
delay(1000); }  
}
```

APÊNDICE B — Código HTML (Página de Login)

```
<!DOCTYPE html>

<html lang="pt-br" >

<head>

<link rel="preconnect"href="https://fonts.googleapis.com" >

<meta charset="utf-8" >

<link rel="stylesheet"type="text/css"href="estilo.css" >

<title>Página de acesso</title>

</head>

<body class="bg" >



<div class="blkcentro" >

<div class="log" >



</div>

<h3>Acesso ao sistema de monitoramento</h3>

<form action="principal.php"method="post"name="validar" > <div class="campos" >

<label><b>Login</b></label>

<input type="text"name="Login"placeholder="Login"required= > <br>

<label><b>Senha</b></label>

<input type="password"name="Senha"placeholder="Senha"size="20"required= > <br>

<button class="btn1"type="submit" >Acessar</button> </div> </form>

</div>

</body>

<footer class="rodape" >

<a href="paginadodesenvolvedor.html" >Desenvolvimento</a>

</footer>

</html>
```

APÊNDICE C — Código HTML (Página Principal)

```
<!DOCTYPE html>

<html lang="pt-br" >

<head>

<link rel="stylesheet"type="text/css"href="estilo.css" >

<title>Banco de Dados</title>

<meta charset="utf-8" >

</head>

<body class="bg" >

<div id="identidade" >



<h1>Acesso ao banco de dados</h1>

<div class="especial" >

<button class="btn2"onclick="window.location.href='consultas.php'" >Fazer consulta

/button> <br><br>

<button class="btn2"onclick="window.location.href='https://phpmyadmin.umbler.com/'

" >Ir ao banco</button></div>

<h2>Inserir abaixo dados ao banco</h2>

<form method="post"action= "process.php" >

<div class="campos2" > <input type="text"name="Identificacao"

placeholder="Identificação"required= >

<input type="text"name="Luminosidade"placeholder="Luminosidade"required= >

<br><br>

<input type="text"name="Temperatura"placeholder="Temperatura"required= >

<input type="text"name="Umidade"placeholder="Umidade"required= >

<p> <input type="submit"Value="Inserir dados" > </p>

</div>

</form> </div>

</body> </html>
```