



BACHAREL EM CIÊNCIA DA COMPUTAÇÃO

**AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO**

VINÍCIUS MAGALHÃES DE QUEIROZ

Rio Verde, GO

2022



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO -
CAMPUS RIO VERDE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO

**AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO**

VINÍCIUS MAGALHÃES DE QUEIROZ

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde, como requisito parcial para a obtenção do Grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Márcio da Silva Vilela
Coorientador: Prof. Rafael Carvalho de Mendonça
Instituto Federal Goiano

Rio Verde, GO

Agosto, 2022

Sistema desenvolvido pelo ICMC/USP
Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas - Instituto Federal Goiano

Q3a Queiroz, Vinícius Magalhães de
AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO / Vinícius Magalhães de Queiroz;
orientador Márcio da Silva Vilela; co-orientador
Rafael Carvalho de Mendonça. -- Rio Verde, 2022.
54 p.

TCC (Graduação em Bacharelado em Ciência da
Computação) -- Instituto Federal Goiano, Campus Rio
Verde, 2022.

1. Automação. 2. Estação meteorológica. 3.
radiofrequência. 4. automática. I. da Silva Vilela,
Márcio, orient. II. Carvalho de Mendonça, Rafael, co-
orient. III. Título.

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610, de 19 de fevereiro de 1998, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano a disponibilizar gratuitamente o documento em formato digital no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

IDENTIFICAÇÃO DA PRODUÇÃO TÉCNICO-CIENTÍFICA

- | | |
|--|---|
| <input type="checkbox"/> Tese (doutorado) | <input type="checkbox"/> Artigo científico |
| <input type="checkbox"/> Dissertação (mestrado) | <input type="checkbox"/> Capítulo de livro |
| <input type="checkbox"/> Monografia (especialização) | <input type="checkbox"/> Livro |
| <input checked="" type="checkbox"/> TCC (graduação) | <input type="checkbox"/> Trabalho apresentado em evento |

Produto técnico e educacional - Tipo:

Nome completo do autor:

Vinicius Magalhães de Queiroz

Matrícula:

2018102201910027

Título do trabalho:

AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO

RESTRIÇÕES DE ACESSO AO DOCUMENTO

Documento confidencial: Não Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: 15 /09 /2022

O documento está sujeito a registro de patente? Sim Não

O documento pode vir a ser publicado como livro? Sim Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O(a) referido(a) autor(a) declara:

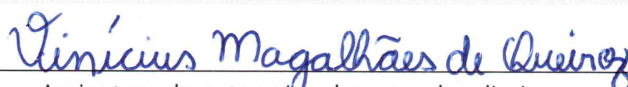
- Que o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- Que obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autoria, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- Que cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Rio Verde

Local

15 /09 /2022

Data



Assinatura do autor e/ou detentor dos direitos autorais

Ciente e de acordo:



Assinatura do(a) orientador(a)



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

Ata nº 59/2022 - GGRAD-RV/DE-RV/CMPRV/IFGOIANO

ATA DE DEFESA DE TRABALHO DE CURSO

Aos 12 dias do mês de agosto, às 13 horas e 30 minutos, reuniu-se a banca examinadora composta pelos docentes: Márcio da Silva Vilela (orientador), Rafael Carvalho de Mendonça (membro) e André da Cunha Ribeiro (membro), para examinar o Trabalho de Curso intitulado "AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO CLIMÁTICO" do estudante Vinícius Magalhães de Queiroz, matrícula nº 2018102201910027 do Curso de Ciência da Computação do IF Goiano – Campus Rio Verde. A palavra foi concedida ao estudante para a apresentação oral do TC, houve arguição do candidato pelos membros da banca examinadora. Após tal etapa, a banca examinadora decidiu pela APROVAÇÃO do estudante. Ao final da sessão pública de defesa, foi lavrada a presente ata que segue assinada pelos membros da Banca Examinadora.

(Assinado Eletronicamente)

Márcio da Silva Vilela

Orientador

(Assinado Eletronicamente)

Rafael Carvalho de Mendonça

Membro

André da Cunha Ribeiro

Membro

Documento assinado eletronicamente por:

- **Andre da Cunha Ribeiro**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 12/08/2022 15:21:57.
- **Rafael Carvalho de Mendonca**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 12/08/2022 15:21:28.
- **Marcio da Silva Vilela**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 12/08/2022 08:36:49.

Este documento foi emitido pelo SUAP em 12/08/2022. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 415093

Código de Autenticação: 74202a0e21

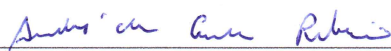


INSTITUTO FEDERAL GOIANO
Campus Rio Verde
Rodovia Sul Goiana, Km 01, Zona Rural, None, None, RIO VERDE / GO, CEP 75901-970
(64) 3620-5600

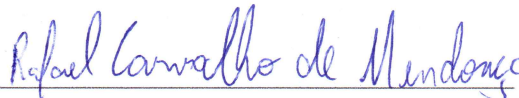
VINÍCIUS MAGALHÃES DE QUEIROZ

**AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO**

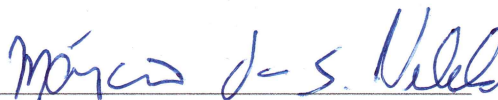
Trabalho de curso DEFENDIDO E APROVADO em 12 de 08 de 2022, pela Banca Examinadora constituída pelos membros:



Prof. Dr. André da Cunha Riberio
Instituto Federal Goiano



Prof. Rafael Carvalho de Mendonça
Instituto Federal Goiano
Coorientador



Prof. Dr. Márcio da Silva Vilela
Orientador

Sistema desenvolvido pelo ICMC/USP
Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas - Instituto Federal Goiano

Q3a Queiroz, Vinícius Magalhães de
AUTOMAÇÃO DE UM SISTEMA DE ARMAZENAMENTO
CLIMÁTICO / Vinícius Magalhães de Queiroz;
orientador Márcio da Silva Vilela; co-orientador
Rafael Carvalho de Mendonça. -- Rio Verde, 2022.
54 p.

TCC (Graduação em Bacharelado em Ciência da
Computação) -- Instituto Federal Goiano, Campus Rio
Verde, 2022.

1. Automação. 2. Estação meteorológica. 3.
radiofrequência. 4. automática. I. da Silva Vilela,
Márcio, orient. II. Carvalho de Mendonça, Rafael, co-
orient. III. Título.

Dedico esse trabalho a minha família, que me deu todo o suporte e apoio para concluir essa longa jornada.

AGRADECIMENTOS

Primeiramente gostaria de agradecer aos meus pais, que dedicaram suas vidas para moldar a pessoa que sou hoje; por sempre apoiarem e incentivarem as minhas escolhas. Agradeço pela orientação nessa trilha complicada da vida.

Aos professores, pelos seus ensinamentos. Em especial, ao professor Márcio da Silva Vilela, sendo meu orientador, ao retribuir dedicação e amizade, ter se tornado um segundo pai para mim. E, ao professor André da Cunha Ribeiro, por ter sido professor, primo e amigo, nas melhores e nas piores horas. Por fim, ao professor Rafael, como meu coorientador, me auxiliando com os protocolos de rede e, nos momentos de saciarmos a fome, principalmente.

“O homem não teria alcançado o possível se, repetidas vezes, não tivesse tentado o impossível.” (Max Weber)

RESUMO

QUEIROZ, Vinícius Magalhães. **Automação de um sistema de armazenamento climático.** Agosto, 2022. 39 f. Monografia – (Curso de Bacharel em Ciência da Computação), Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde. Rio Verde, GO, Agosto, 2022.

As alterações climáticas estão se intensificando ao longo dos anos devido a fatores ainda não totalmente esclarecidos pela comunidade científica. Para reduzir os impactos da escassez de água, os agricultores tem recorrido, frequentemente, ao manejo da água de irrigação, por meio de cálculos de evapotranspiração. Para realização destes cálculos é necessário o uso de dados climáticos de estações meteorológicas próximas ao local, que possam rapidamente disponibilizá-los. Para atender esta necessidade, este trabalho propõe transformar uma estação meteorológica semiautomática em automática, por meio da modificação de seu software e a inserção do transmissor de radiofrequência, permitindo que os dados possam ser enviados até um receptor conectado à rede de internet. Deste modo, gera a visualização dos dados de forma remota e sua transferência para um banco de dados web. O banco de dados em questão, faz parte de um programa para o cálculo da evapotranspiração, que foi modificado para se ajustar ao processo proposto. Os resultados são apresentados neste trabalho.

Palavras-chave: Estação meteorológica, automática, atualização, radiofrequência.

ABSTRACT

QUEIROZ, Vinícius Magalhães. Automation of a climate storage system. Agosto, 2022. 39 f. Trabalho de Conclusão de Curso – Bacharel em Ciência da Computação, Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde. Rio Verde, GO, Agosto, 2022.

Climate change is intensifying over the years due to factors not yet fully understood by the scientific community. To reduce the impacts of water scarcity, farmers have often resorted to managing irrigation water through evapotranspiration calculations. To carry out these calculations, it is necessary to use climate data from meteorological stations close to the site, which can quickly make this data available. To meet this need, this work proposes to transform a semi-automatic meteorological station into an automatic one, through the modification of its software and the insertion of the radiofrequency transmitter, allowing the data to be sent to a receiver connected to the internet network. In this way, it generates the visualization of data remotely and its transfer to a web database. The database in question is part of a program for calculating evapotranspiration, which was modified to fit the proposed process. The results are presented in this work.

Keywords: Weather station, automatic, update, radio frequency.

LISTA DE FIGURAS

Figura 1 – Estações Meteorológicas do Brasil (INMET, 2022)	4
Figura 2 – Arduino nano (ARDUINO, 2022b)	10
Figura 3 – Sensor NRF24L01+ com sua antena	11
Figura 4 – Esboço da montagem do circuito	11
Figura 5 – Diagrama EMA para desenvolvimento do menu	14
Figura 6 – Placa de circuito impresso desenvolvida no software Kicad para fabricação	16
Figura 7 – Bancada de teste para programação do transceptor	17
Figura 8 – Código desenvolvido para a banca de transmissão e recepção	18
Figura 9 – Dados de teste recebidos pela banca do receptor	18
Figura 10 – Estouro de memória do arduino uno usando as diretivas de banco de dados	19
Figura 11 – Nova tabela no banco de dados	20
Figura 12 – Trigger desenvolvida para inserir os dados nas tabelas principais	20
Figura 13 – Esquema do sistema de comunicação da estação meteorológica com o servidor.	21
Figura 14 – Instalação do módulo transmissor	22
Figura 15 – Controlador PIC18F480 da estação meteorológica	23
Figura 16 – Circuito de teste para simulação dos dados da estação	23
Figura 17 – Local do teste de alcance do transceptor	24
Figura 18 – Placa de circuito impresso fabricada	25
Figura 19 – Receptor montado	25
Figura 20 – Site hospedado no servidor do IFGoiano - Campus Rio Verde	26

LISTA DE TABELAS

Tabela 1 – Tabelas de materiais utilizados para a construção do receptor	15
--	----

LISTA DE ABREVIATURAS E SIGLAS

KB	Kilobyte
RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)
IP	<i>Internet Protocol</i> (Protocolo de Internet)
TCP	<i>Transmission Control Protocol</i> (Protocolo de Controle de Transmissão)
FAO	Organização das Nações Unidas para Alimentação e Agricultura
INMET	Instituto Nacional de Meteorologia
LCD	<i>Liquid Crystal Display</i> (Tela de Cristal Líquido)
EMA	Estação Meteorológica Automática
EMA-BC	Estação Meteorológica Automática de Baixo Custo
MHz	Megahertz
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
LAN	<i>Local Area Network</i> (Rede de área local)
Mbps	Megabit por segundo
GHz	Gigahertz
LED	<i>Light Emitting Diode</i> (Diodo Emissor de Luz)
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
mm	Milímetro
mA	Miliampere
SPI	<i>Serial Peripheral Interface</i> (Interface Periférica Serial)
TX	Trasmitir
VIN	Entrada de energia
GND	Ground (Aterramento)
EEPROM	<i>Electrically Erasable Programmable Read-Only Memory</i> (Memória Somente de Leitura Programável Apagável Eletricamente)

SD	<i>Secure Digital Card</i>
SQL	<i>Standard Query Language</i>
V	Volts
PIC	<i>Programmable Interface Controller</i>
W	Watts
GB	Gigabytes

SUMÁRIO

1 – INTRODUÇÃO	1
2 – Revisão da Literatura	3
2.1 Meteorologia	3
2.2 Instituto Nacional de Meteorologia	3
2.3 Estações meteorológicas	4
2.3.1 Estações meteorológicas convencionais (EMCs)	4
2.3.1.1 Estação convencional	5
2.3.1.2 Estação semiautomática	5
2.3.2 Estação meteorológica automáticas (EMAs)	5
2.3.2.1 Formas de Comunicação	6
2.3.2.1.1 Cabeamento	6
2.3.2.1.2 Rede sem fio	6
2.4 Evapotranspiração	8
2.4.1 Como é calculado	8
2.5 Arduino	9
3 – Materiais e Métodos	10
3.1 Modificação da estação meteorológica	10
3.2 Receptor de dados	13
3.3 Servidor e banco de dados	16
4 – Desenvolvimento	17
4.1 Transmissão	17
4.2 Receptor	19
4.3 Banco de dados	20
4.4 Sistema geral	21
5 – Resultados e Discussão	22
5.1 Alteração da estação	22
5.2 Receptor	25
5.3 Banco de dados	26
5.4 Servidor	26
6 – Conclusão	27
Referências	29

Anexos	31
ANEXO A –Código fonte do transmissor	32
ANEXO B –Código fonte do receptor	35

1 INTRODUÇÃO

Hodiernamente, o planeta tem passado por grandes mudanças climáticas devido ao efeito estufa, causado por desmatamentos e queima de combustíveis fósseis. Estas mudanças têm afetado de sobremaneira a agricultura, reduzindo a incidência de chuvas em certas regiões e aumentando em outras.

Cada vez mais é necessário utilizar a agricultura irrigada para garantir a produção de grãos, evitando possíveis perdas, devido à falta de água. Ao mesmo tempo, o uso da água deve ser feito de forma controlada, para coibir desperdícios nos recursos hídricos.

Silva e Vale (2007) afirmam que para o gerenciamento da irrigação, é necessário a implementação de medidas e procedimentos que buscam saber quanto e quando irrigar. Um dos procedimentos mais usados para cumprir com este objetivo, consiste no cálculo da evapotranspiração pelo método Penman Moteith, recomendado pela FAO (Organização das Nações Unidas para Alimentação e Agricultura). Este método utiliza dados climáticos sobre a radiação solar global, velocidade do vento, temperatura e umidade, além da taxa de precipitação pluviométrica.

Em todo o globo há uma expansão do uso da programação em conjunto com a automação para a coleta de dados, armazenamento e disseminação de informações. A área da meteorologia é uma delas, por meio das estações meteorológicas, torna-se possível estimar o clima e realizar previsões para diversos tipos de culturas na agricultura. (SILVA et al., 2018).

Há dois tipos básicos de estação meteorológica: as automáticas e as convencionais. As estações automáticas realizam a transferência dos dados coletados para um banco de dados e disponibilizam as informações ao usuário, de forma automática e em tempo real. As estações convencionais necessitam que o usuário faça a transferência de dados de forma manual, lendo os dados de um datalogger ou a partir de uma unidade de armazenamento (INMET, 2022).

Uma estação meteorológica automática é um sistema autônomo de mensuração e registro dos dados de fenômenos meteorológicos. Os sensores usados para calcular a evapotranspiração com base na metodologia FAO são: anemômetro, higrômetro, piranômetro e termômetro. Finholdt et al. (2011) definem sensor como um dispositivo sensível à alguma forma de energia, relacionando-a com uma grandeza física.

Segundo Finholdt et al. (2011) o maior acesso às estações meteorológicas automáticas, permitirá a implantação de programas de gerenciamento de irrigação mais eficientes, reduzindo o consumo de água, energia elétrica, mão de obra, menor ocorrência de doenças em plantas e maior eficiência no uso de produtos fitossanitários.

As grandezas físicas (irradiância solar, temperatura do ar e umidade relativa do ar, entre outras) registradas em sistemas de aquisição de dados são inerentes a alguns

fenômenos físicos, e, em geral, sua natureza é analógica. Uma variável analógica assume valores contínuos e reais, diferentemente dos sinais digitais, que são descontínuos e expressos em uma representação binária. As grandezas analógicas podem ser obtidas por meio de sensores eletrônicos. Assim, um sensor eletrônico (estado sólido) é um dispositivo que cria um campo elétrico, induz uma tensão ou uma corrente e reage a qualquer alteração desse campo elétrico (FINHOLDT et al., 2011).

2 Revisão da Literatura

2.1 Meteorologia

“A meteorologia é definida como a ciência que estuda os fenômenos que ocorrem na atmosfera, e está relacionada ao estado físico, dinâmico e químico da atmosfera, as interações entre elas e a superfície terrestre subjacente”. Isto é, a meteorologia é o estudo dos fenômenos climáticos em um determinado período de tempo podendo variar conforme se investiga. (INMET, 2022).

Para Ynoue et al. (2017) meteorologia é a ciência que estuda processos atmosféricos e como estão interligados com o sistema terrestre.

A agricultura é dependente das condições climáticas para seu desenvolvimento, pois depende de uma série de fatores como temperatura, umidade, intensidade do sol e, principalmente, água para sua sobrevivência. Dessa forma, agricultura e meteorologia realizam um mutualismo entre si. A meteorologia para agricultura é essencial, pois possibilita que o produtor faça planejamentos estratégicos para trabalhar da melhor forma (RIBEIRO; SIMEÃO; SANTOS, 2016).

Nesse sentido é, portanto, um fator determinante no que se refere à maneira como será a produção. É por meio dela que as atividades são planejadas no tempo exato para que se possa aproveitar ao máximo o clima, de modo que ele traga apenas vantagens e benefícios (RIBEIRO; SIMEÃO; SANTOS, 2016; NASUTI et al., 2013).

2.2 Instituto Nacional de Meteorologia

O Instituto Nacional de Meteorologia (INMET) é um órgão do Ministério da Agricultura, Pecuária e Abastecimento. Sua principal função no país é fornecer informações meteorológicas à população brasileira, ajudando no processo de tomada de decisões por meio do monitoramento, análise e previsão climática (INMET, 2022).

O sistema de coleta de dados do INMET realiza a aquisição dos dados dos seguintes índices climáticos: temperatura, umidade relativa do ar, direção e velocidade do vento, pressão atmosférica, precipitação, entre outras variáveis. Possui estações de sondagem superior (radiossonda), estações meteorológicas de superfície, operadas manualmente e, a maior rede de estações automáticas da América do Sul. Todos os seus dados são disponibilizados de forma bruta e gratuita, sem nenhum tipo de tratamento a base de cálculos sobre eles (INMET, 2022).

O INMET realiza o cadastro de estações meteorológicas em seu sistema classificando-as para diferenciar das cadastradas. Sua representação no mapa da figura 1 mostra que há três tipos de estações: representadas por verde são as meteorológicas automáticas, as em azul, são as convencionais e as roxas são denominadas pluviométricas (INMET, 2022).

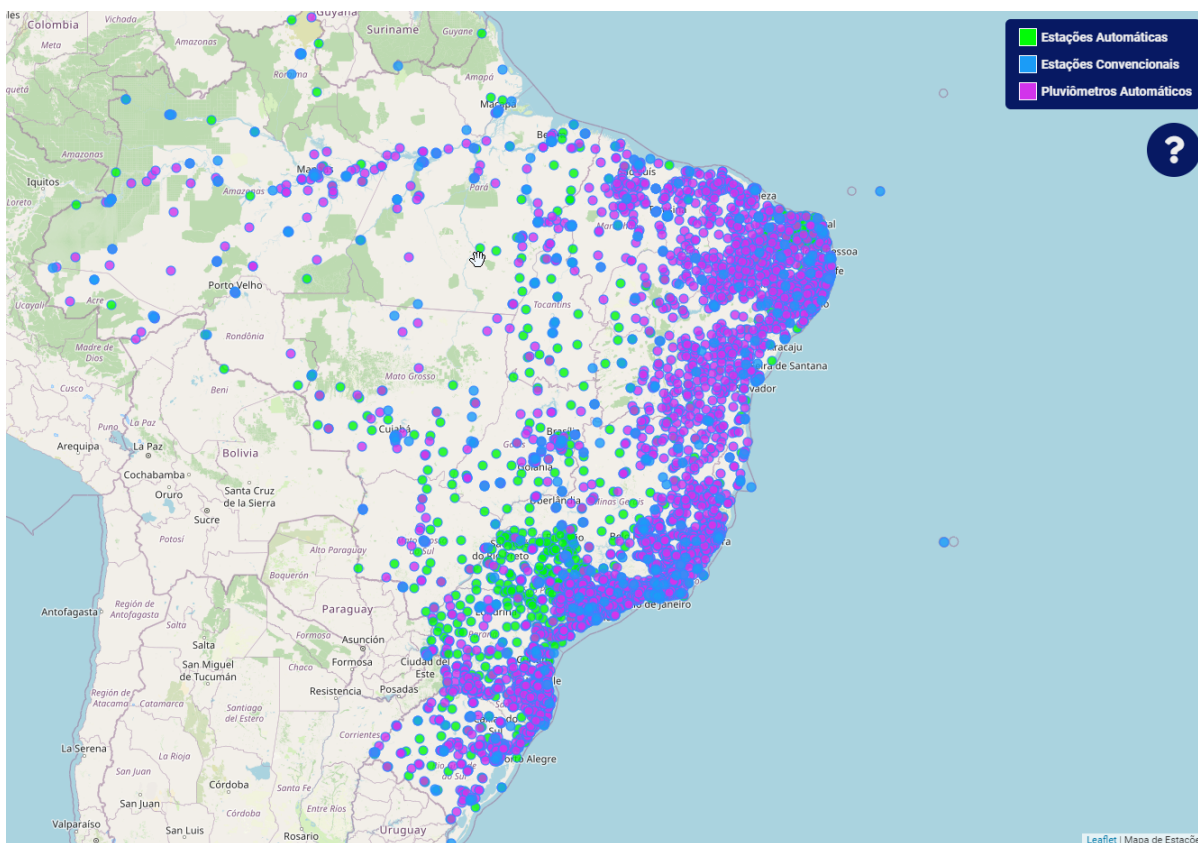


Figura 1 – Estações Meteorológicas do Brasil (INMET, 2022)

Os dados coletados das estações são disponibilizados de maneira bruta, ou seja, sem nenhuma análise prévia apenas dados em si. Em alguns casos os dados são tratados chegando a resultados que são apresentados em seu site, como previsões de chuva, vento, intensidade do sol, entre outros.

2.3 Estações meteorológicas

Uma estação meteorológica tem como função medir fenômenos atmosféricos, é dotada com instrumentos e sensores que realizam a conversão dos fenômenos físicos em dados numéricos a serem computados. Seus principais sensores são de temperatura, pressão atmosférica, umidade, velocidade dos ventos, direção solar e precipitação (PERAZZI et al., 2021).

2.3.1 Estações meteorológicas convencionais (EMCs)

O nome técnico deste tipo de estação é Estação Meteorológica de Observação de Superfície Convencional. Uma estação meteorológica convencional, consiste em diferentes sensores isolados que medem continuamente parâmetros meteorológicos (pressão atmosférica, temperatura e umidade relativa do ar, precipitação, radiação). Não possui nenhuma forma de comunicação com o meio externo, assim, havendo a necessidade de uma

intervenção de um usuário para coletar os dados manualmente (INMET, 2022; SOUZA, 2018).

Esse tipo de estação pode ser dividido em duas classes:

2.3.1.1 Estação convencional

As estações convencionais realizam coletas dos dados como qualquer outra estação, porém elas não armazenam essas informações em nenhum lugar. A forma de visualização desse modelo é por um display LCD que, em sua maioria, fica acoplado junto ao hardware da estação. Os dados lidos pelos sensores são coletados pelo operador e inseridos em uma planilha, manualmente.

2.3.1.2 Estação semiautomática

As convencionais e semiautomáticas realizam o mesmo trabalho de leitura dos sensores instalados, no entanto, existe uma diferença significativa entre elas. A diferença está na forma em que os dados são armazenados. Enquanto as convencionais não armazenam nada, as semiautomáticas possuem um temporizador que após um determinado período, armazena os dados em uma memória externa. Em seguida, o operador ainda precisa pegar esses dados e transferi-los para uma planilha.

2.3.2 Estação meteorológica automáticas (EMAs)

Uma Estação Meteorológica Automática (EMA) é todo conjunto de sensores que se captam os dados e transmitem automaticamente a outro sistema sem a intervenção humana. Sua utilização é dada quando se quer aumentar a confiabilidade nos dados (PERAZZI et al., 2021).

Segundo Organization (2021) existem quatro (4) categorias que as estações podem ser divididas:

1. Estações Light (mínimas) para medição de poucas variáveis como temperatura e/ou precipitação;
2. Estações básicas para medições meteorológicas básicas (temperatura do ar, umidade relativa, velocidade e direção do vento, precipitação e pressão atmosférica);
3. Estações estendidas com o intuito de medir adicionalmente radiação solar, a duração da luz do sol, a temperatura do solo e a evaporação;
4. Estações completas, com todos os dados anteriores, acrescentando a medição da altura entre a base da nuvem e o clima atual;

Todas as categorias fornecem a capacidade de registrar dados usando dados proprietários, bem como a capacidade de transmitir dados usando uma variedade de métodos. Adicionalmente, as categorias mencionadas de EMA, Organization (2021) reconhece outro tipo de estação climática brevemente intitulada como Estação Meteorológica Automática -

Baixo Custo (EMA-BC). Esse tipo de estação é caracterizada por seu baixo custo de uso e compra, bem como o baixo consumo de energia, a capacidade de transmissão de dados em tempo real (com ou sem registro), e finalmente seu tamanho, que é pequeno e compacto (IOANNOU et al., 2021).

A estação utilizada para esta pesquisa é a estação do tipo estendida, que se encontra atualmente no Laboratório de Automação e Controle do Instituto Federal Goiano - Campus Rio Verde.

2.3.2.1 Formas de Comunicação

Quando fala-se de estações automáticas, logo entende-se que não há a necessidade de intervenção humana no equipamento para coletar os dados. No entanto, para que os dados possam ser enviados aos sistemas é necessário criar uma ponte entre eles. Para isso, utiliza-se equipamentos de comunicação por fios ou por ondas eletromagnéticas. Tais tipos de comunicação são descritas como:

Tais tipos de comunicação são descritas como:

2.3.2.1.1 Cabeamento

Neste modo de envio, usa-se fios de condutibilidade elétrica que trafegam dados por meio da eletricidade. São popularmente conhecidos como cabos de internet (Cabo de par trançado), cabo de TV e câmeras (Cabos coaxiais). fibra óptica e fios comuns de cobre. Todos esses tipos de materiais são possíveis para enviar dados, porém possuem limitações individuais (PINHEIRO, 2016).

Segundo Pinheiro (2016) as limitações são:

- **Cabo de par trançado** - possui alta velocidade de comunicação, porém limita-se a 100 metros de distância. Após essa distância, há perda de dados na comunicação.
- **Cabo Coaxial** - Possui três camadas, a primeira é uma camada externa de plástico, a segunda uma malha de blindagem e o dielétrico como condutor central. Sem repetidores de sinal, o seu alcance máximo é de 300 metros.
- **Fibra óptica** - Possui alcance de 2.000 metros, valor alto devido a qualidade e construção do material, equipamentos de conversão de informação de alto valor e não é maleável. Ela pode sofrer rompimento com facilidade devido seu material possuir vidro.

2.3.2.1.2 Rede sem fio

Os dispositivos sem fio estão em toda parte. Telefones celulares são itens de consumo muito comuns. Ademais, existe uma tendência em substituir o cabeamento das redes de internet por redes sem fio. A introdução desses serviços aumenta a mobilidade e a

área de atuação de muitos projetos existentes, sendo um deles as estações meteorológicas automáticas (HAYKIN; MOHER, 2008).

Diversas redes sem fio podem ser usadas nas estações meteorológicas tais como:

1. **Bluetooth**

Também conhecido como padrão 802.15.1 da IEEE, foi desenvolvido como um sistema de comunicação sem fio para substituir os cabos de pequena extensão para equipamentos portáteis. Opera na faixa de frequência de 2.400 Mhz a 2.483,5 Mhz. Sua distância máxima de transmissão de dados sem nenhum obstáculo é de 10 metros.

2. **Wifi**

Padrão LAN sem fio denominado 802.11, popularmente conhecido como Wifi. Sua frequência operacional fica em torno de 2.4 a 2.5 GHz e 5.725 a 5.825 GHz. As bandas de operação possuem características semelhantes como envio de dados por ondas eletromagnéticas, porém possuem diferenças em alcance e velocidade. A rede 2.4 GHz possui um alcance médio de 250 metros em áreas externas, podendo transmitir taxas de dados de até 125 Mbps. A rede 5 GHz possui um alcance menor em áreas externas de 70 metros, mas podendo transmitir mais de 500 Mbps de velocidade de banda (HAYKIN; MOHER, 2008; TANENBAUM, 2011; OLIVEIRA; ALENCAR; LOPES, 2018).

3. **Rede móvel**

Segundo Oliveira, Alencar e Lopes (2018) e as pessoas adoram falar ao telefone e ainda mais navegar pela Internet e, isso tem tornado a rede de telefonia móvel a mais bem-sucedida do mundo.

- Os sistemas de telefonia móvel de primeira geração transmitiam chamadas de voz como sinais com variação contínua (analógicos) ao invés de sequências de bits (digitais).
- Os sistemas de telefonia móvel de segunda geração comutavam para transmitir a capacidade, melhorar a segurança e oferecer mensagens de texto. Tornou-se o sistema mais usado do mundo.
- Os sistemas de terceira geração, ou 3G, foram implantados inicialmente em 2001 e oferecem tanto voz digital como serviços de dados digitais de banda larga. Oferece 14 Mbps de download e 6 Mbps de upload.
- Devido a escassez de espectros de banda para propagação das ondas de telefonia móvel gerou-se um projeto chamado de rede celular. O intuito desse projeto é reutilizar as frequências em redes vizinhas para aumentar a capacidade da rede.

Devido ao grande aumento do número de aparelhos conectados, as empresas de telefonia móvel, não conseguem acompanhar na mesma proporção o aumento da quantidade de aparelhos conectados por antena em área de pouca habitação. A zona rural é a mais afetada por essa escassez de sinal, isso porque se a região for muito afastada da cidade o sinal não conseguem chegar ao aparelho e ser decodificado.

Antenas de telefones são utilizadas para contornar tal situação, mas ainda não consegue fornecer a mesma qualidade de banda estipulada pela operadora (MILANEZ et al., 2020; OLIVEIRA; ALENCAR; LOPES, 2018).

2.4 Evapotranspiração

O conceito de evapotranspiração, o mais significativo avanço no conhecimento dos aspectos da umidade climática, foi introduzido em 1948, por **Thorntwaite e Parmen**. Desde então, os modelos empíricos e físicos para a estimativa de ETo com base no monitoramento de elementos meteorológicos evoluíram muito, com isso, padronizou-se uma equação a qual é denominada Penman-Monteith, parametrizada pela FAO. A referência de evapotranspiração (ETo), representa a demanda de água de uma cultura de acordo com os elementos climáticos de um determinado local. Além disso, é um parâmetro de extrema importância para a definição das necessidades hídricas das lavouras (LEITE et al., 2020; INMET, 2022).

A agricultura tem uma grande relação com a evapotranspiração, pois utiliza-se do cálculo para estimar a quantidade de água necessária para repor ao solo. Olhando um ecossistema mais a fundo, pode-se observar que as plantas realizam um processo chamado de transpiração. Nesse caso, afirmamos ser um processo no qual a planta libera água no estado gasoso por meio de seus poros. Como a planta perde muita água, há uma necessidade de reposição, a qual será retirada do solo por meio das raízes.

2.4.1 Como é calculado

Para a realização do cálculo através do método de Penman-Monteith FAO, usam-se os dados de temperatura, umidade relativa, velocidade dos ventos e radiação solar. O modelo descrito por este método é dado pela fórmula abaixo

$$ET_o = \frac{0,408\Delta(R_n - G) + \gamma \frac{900}{T+273} u^2 (e_s - e_a)}{\Delta + \gamma(1 + 0,34u^2)} \quad (1)$$

Em que,

- ETo - evapotranspiração de referência (mmd^{-2});
- R_n - saldo de radiação à superfície da cultura ($MJm^{-2}d^{-1}$);
- G - densidade do fluxo de calor do solo ($MJm^{-2}d^{-1}$);
- T - temperatura média do ar ($^{\circ}C$);
- u_2 - velocidade média do vento a 2m de altura (ms^{-1});
- e_s - pressão de vapor de saturação para temperatura média (kPa);
- e_a - pressão parcial de vapor (kPa);
- Δ - declividade da curva de pressão de vapor no ponto de temperatura média ($kPa^{\circ}C^{-1}$);
- γ - coeficiente psicrométrico ($kPa^{\circ}C^{-1}$);

2.5 Arduino

O Arduino é uma plataforma de prototipagem de código aberto baseada em hardware e software fáceis de usar. As placas Arduino são capazes de ler entradas de luz em um sensor, um dedo em um botão ou uma mensagem do Twitter e transformá-la em uma saída, ativando um motor, ligando um LED, publicando algo online. Para isso, basta utilizar a linguagem de programação do Arduino (baseada em Wiring), e o Software Arduino(IDE) (ARDUINO, 2022b).

A vantagens da utilização do arduino dá-se pelos seguintes fatores:

1. **Multiplataforma** - Por ser um software de código aberto, seu sistema o Arduino(IDE) é executado em diferentes sistemas operacionais como Windows, MacOS e Linux. Ademais, outros sistemas de microcontroladores são limitados apenas ao Windows.
2. **Ambiente de programação simples** - O software Arduino(IDE) é fácil de utilizar, pois possui flexibilidade no sistema dando ao usuário funções claras e de fácil acesso.
3. **Código aberto e extensível** - A linguagem pode ser expandida por meio de bibliotecas C, e as pessoas que desejam entender os detalhes técnicos podem dar o salto do Arduino para a linguagem de programação AVR-C, na qual se baseia. Da mesma forma, você pode adicionar o código AVR-C diretamente em seus programas Arduino, se desejar.

3 Materiais e Métodos

3.1 Modificação da estação meteorológica

A base para o início do projeto foi o estudo da estação meteorológica em uso no IFGoiano - Campus Rio Verde, onde o modelo da estação se encontrava no Laboratório de Automação, Simulação e Controle desenvolvido pelo Santos (2022b). O modelo inclui um sensor de velocidade do vento, radiação solar, pluviometria, temperatura e umidade. Com base nesse projeto, foram pesquisadas formas de ler esses dados diretamente da estação e enviá-los por meio de ondas de rádio a um receptor.

O projeto não visa recriar a estação meteorológica e sim aprimorá-la. Para isso, havia a necessidade de usar outro microcontrolador. Diversos microcontroladores foram estudados com a finalidade de que possuíssem um baixo consumo de energia, interface de programação intuitiva, dimensão pequena e uma quantidade de portas digitais e analógicas, satisfatórias para leitura das informações na estação.

O arduino nano, mostrado na figura 2, foi o que apresentou melhor compatibilidade com os requisitos do projeto, tendo uma interface de fácil programação usando sua própria IDE, baixo consumo de energia, chegando a 19 mA, tamanho de 18 x 45 mm e, é construído com 13 portas digitais e 7 portas analógicas ao todo (ARDUINO, 2022a).



Figura 2 – Arduino nano (ARDUINO, 2022b)

Dentre os métodos que poderiam ser utilizados, a transmissão por rádio frequência foi o que melhor se ajustou aos requisitos desejados de baixo consumo, alcance e independência de uma infraestrutura das redes de telefonia, pois as redes móveis como 4G e 5G, não estão disponíveis em todos os locais onde se pratica a agricultura irrigada (HAYKIN; MOHER, 2008; TANENBAUM, 2011).

Para a transmissão por radiofrequência, foi utilizado o transceptor NRF24L01+, mostrado pela figura 3, que opera dentro da faixa de 2,4 e 2,5 GHz. Sua potência de saída, canais de frequência e configuração de protocolo são programáveis por uma interface SPI. O consumo de corrente é de 12,3 mA no modo TX e permite que o módulo se mantenha modo *Built-in Power Down* para economia de energia (ENGINEERS, 2022).

Com uma antena de alcance de até 1 Km, o que atende a especificação de alcance desejada no projeto. Para um alcance maior, seria necessário utilizar uma antena direcional

ou fazer uso de um transmissor mais potente, o que poderia exceder a faixa de energia disponível na estação, produzida por uma placa fotovoltaica de 10 W (SANTOS, 2022b; ENGINEERS, 2022).

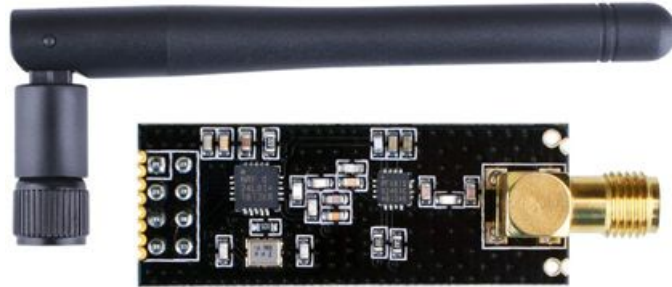


Figura 3 – Sensor NRF24L01+ com sua antena

Para a programação do arduino a fim de estabelecer leitura e envio dos dados, foi utilizado o Arduino IDE. A linguagem de programação C e o uso das bibliotecas do transceptor NRF24L01+ foram usadas para enviar os dados ao receptor. Para facilitar, o programa foi estruturado em pseudocódigo que é mostrado no algoritmo 1. Nele, há a apresentação da lógica geral de como os dados serão lidos e enviados.

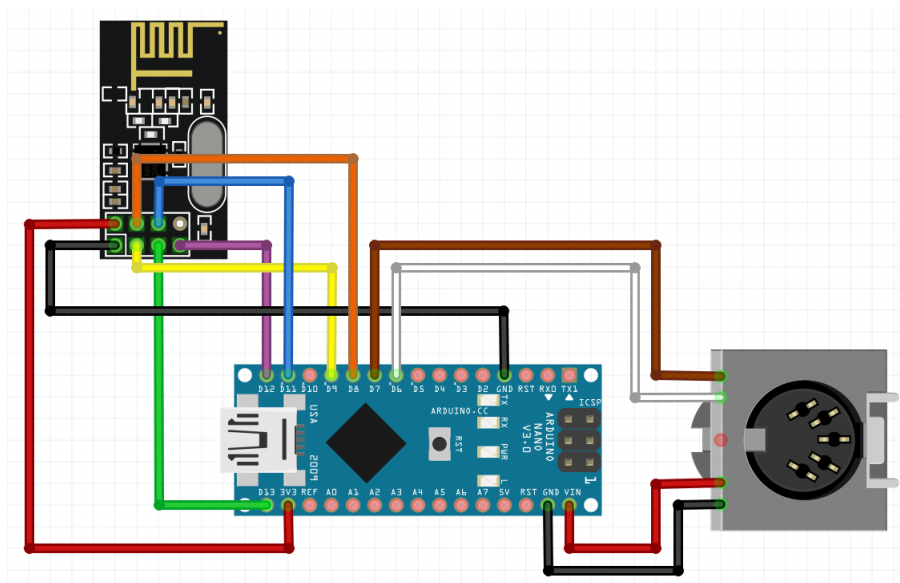


Figura 4 – Esboço da montagem do circuito

Com o levantamento de todos os componentes e suas dependências utilizou-se o software Fritzing como um método de visualizar como seria a montagem do circuito. A figura 4, demonstra a montagem do circuito completa, pode-se ver que temos um conector de 5 pinos fêmea, que será por onde o transmissor receberá a energia para o funcionamento. O arduino nano recebe essa energia pelas portas VIN e GND, os dados da estação são lidos pelas portas digitais 7 e 6. O transceptor NRF24L01+ é conectado ao arduino usando o padrão de protocolo SPI, assim como o da estação, para o envio dos dados.

Algoritmo 1: Algoritmo de envio de dados da estação meteorológica**Entrada:** Vetor de caracteres Dados contendo os dados meteorológicos**Saída:** Média aritmética de cada índice climático

```

1 início
2   Dados[80] ← Vazio;
3   enquanto (Porta6 = Falso E Porta7 = Falso) faça
4     |   Dados[0] ← Porta5;
5     |   K ← K + 1
6   fim
7   para n < 80 faça
8     |   se Porta7 = Falso então
9     |     |   Pare
10    |     fim
11    |     Dados[n] ← Porta5;
12    |     K ← K + 1;
13  fim
14  Aguarde 0,3 segundos;
15  Temperatura ← 0;
16  Umidade ← 0;
17  Velocidade ← 0;
18  Radiação ← 0;
19  Pluviometria ← 0;
20  se n = 80 então
21    |   para n de 15 até 0 faça
22    |     |   Temperatura ← 1+dados[n]
23    |     fim
24    |     para n de 15 até 0 faça
25    |     |   Umidade ← 1+dados[n+16]
26    |     fim
27    |     para n de 15 até 0 faça
28    |     |   Velocidade ← 1+dados[n+32]
29    |     fim
30    |     para n de 15 até 0 faça
31    |     |   Pluviometria ← 1+dados[n+48]
32    |     fim
33    |     para n de 15 até 0 faça
34    |     |   Radiação ← 1+dados[n+64]
35    |     fim
36  fim
37 fim

```

Como a estação possui recursos energéticos limitados, o transceptor não poderia enviar os dados na medida que é lido na estação. A lógica implementada nesta situação, retratada pelo algoritmo 1, foi pegar todos os dados lidos pela estação em um período de 5 minutos, calcular a média aritmética e enviar o resultado dessa média, referente a cada tipo de dado coletado. A lógica implica em acionar apenas uma única vez o transceptor a

cada 5 minutos, poupando a energia da estação.

3.2 Receptor de dados

Os requisitos necessários para desenvolvimento do receptor baseou-se em um controlador que pudesse operar com o mesmo transceptor usado no transmissor. Ter capacidade de memória para comportar a programação de um menu e protocolos de comunicação. Comportar conexão via internet, pois através dela seriam enviados os dados recebidos para o banco de dados, via protocolo TCP/IP.

O arduino Uno mostrou melhor compatibilidade com os requisitos. Manteve-se o mesmo chip de processamento, pois tanto o arduino nano quanto o uno, utilizam o mesmo controlador o ATmega328. O módulo possui pinagens fêmeas de conexão possibilitando o acoplamento de módulos externos já prontos ou fabricados para uma função específica. Ele conta com 32 KB de Flash (mas 512 Bytes são utilizados pro bootloader), 2 KB de RAM e 1 KB de EEPROM, essa quantidade atende aos requisitos de desenvolvimento do código fonte (ARDUINO, 2022b).

Para acesso à internet, o módulo Ethernet apresentou melhor compatibilidade com os requisitos levantados. Conecta a rede de internet via cabo de rede, recebendo um IP e podendo trabalhar com o sistema de protocolo TCP/IP. Sua construção possui pinos machos na parte inferior, permitindo encaixar em cima do arduino uno. Caso haja necessidade de armazenar informações extras, permite trabalhar com cartões SD. Como é conectado na parte superior do arduino, o módulo também reutiliza as mesmas conexões do arduino uno em seu próprio corpo.

Como o sistema receberá instruções do usuário, um display LCD 20x4 e três botões de pulso, foram necessários. O display terá a finalidade de apresentar o menu de configurações implementado e para que o usuário possa configurar o receptor, três botões de pulso foram colocados permitindo a interação com o menu.

O diagrama EMA foi usado para organizar as ideias e definir instruções que serão apresentadas ao usuário por meio do menu. A figura 5 representa como será a interação do usuário com o receptor e quais decisões tomará, visualizando as informações no display LCD.

Como o sistema é um receptor, o melhor componente a ser usado seria o mesmo do transmissor o transceptor NRF24L01+, pois além de enviar dados, seu circuito permite recebê-los. Esse transceptor possibilita operar em múltiplos canais, permitindo que muitas comunicações sejam feitas sem causar interferência entre elas. Quando configurado como receptor, tem a capacidade de ouvir até seis transmissores simultaneamente. Isto significa que podemos unificar a comunicação em uma configuração de teia de modo que qualquer módulo integrante da rede possa fornecer ou adquirir dados de qualquer outro membro. O módulo possui oito pinos, e se comunica com o microcontrolador pelo protocolo SPI (SEMICONDUCTOR, 2008).

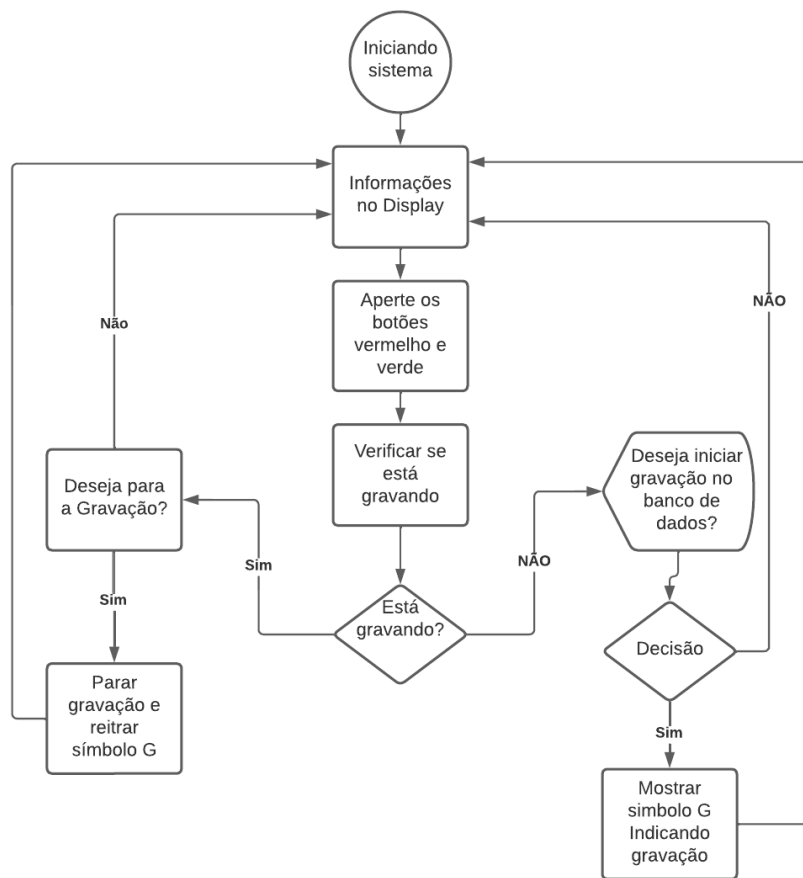


Figura 5 – Diagrama EMA para desenvolvimento do menu

Os reguladores de tensão foram utilizados juntos ao sensor NRF24L01+, pois para enviar e receber informações ele precisa de uma tensão estável, a qual a porta digital do arduino não consegue manter. Assim, o adaptador regula a interferência da corrente elétrica utilizando um capacitor de $10\mu f$, melhorando o alcance do sensor e diminuindo a perda de pacotes de dados (ENGINEERS, 2022).

A tabela 1 representa todos os componentes descritos nesse tópico.

Como o módulo Ethernet apresentado na tabela 1, possui portas para inserção de pinos de comunicação, uma placa de circuito impresso foi proposta para conectar todos os componentes ao arduino. A utilização da placa de circuito impresso diminui os riscos de mal contato entre os componentes.

Para a confecção da placa de circuito impresso analisou-se todas as pinagens de cada componente usado no projeto e como todos seriam interligados. Antes de mandar para a fabricação a modelagem era necessária. O software Kicad foi usado para o desenvolvimento do diagrama de fabricação, como mostra a figura 6, por ser um software gratuito e de fácil manipulação.

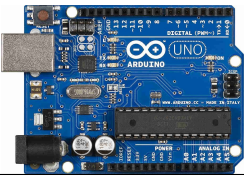


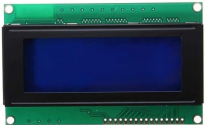

Imagem do componente	Nome do componente	Função
	Arduino Uno	Realiza todo o controle de receber os dados e enviá-los pela rede
	Sensor de radiofrequência NRF24L01+	Recebe os dados da estação via ondas de rádio
	Regulador de Tensão e Corrente	Usa a tensão e corrente do arduino e estabiliza para que o módulo NRF24L01+ possa funcionar corretamente.
	Botões de pulso	Usado para realizar a interação entre o usuário e a interface do receptor
	Tela LCD 20x4	Realiza a visualização das informações do receptor e configuração do mesmo
	Caixa de Patola	Armazena toda a montagem do circuito elétrico, pois possui propriedades antiestáticas.
	Módulo Ethernet	Transfere todas as informações coletadas pelo arduino e envia diretamente para o banco de dados por meio de protocolo IP.
	Fios de Cobre maleáveis	Conecta todos os componentes na placa de circuito impresso

Tabela 1 – Tabelas de materiais utilizados para a construção do receptor

Para a armazenagem dos componentes foi utilizada uma caixa de patola. A mesma possui propriedades antiestáticas e cobre todo o controlador de agentes externos.

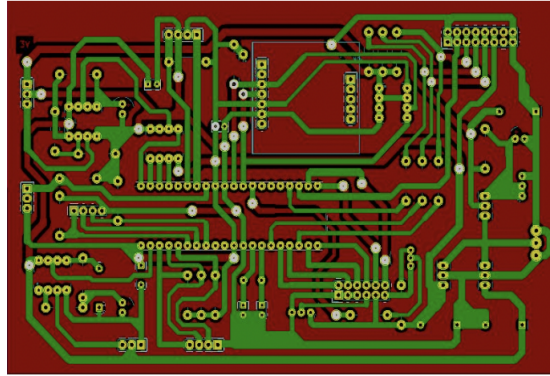


Figura 6 – Placa de circuito impresso desenvolvida no software Kicad para fabricação

3.3 Servidor e banco de dados

Para a hospedagem e armazenamento da aplicação desenvolvida por Santos (2022a) verificou-se os seguintes requisitos. Servidor que possui memória de armazenamento igual ou superior a 50 gigabytes, aceite requisição do navegador e um banco de dados localmente.

Devido a esses requisitos, o Instituto Federal Goiano - Campus Rio Verde forneceu um espaço para armazenamento de todo o sistema, no servidor do bloco da computação. Esse equipamento atende perfeitamente aos requisitos do software desenvolvido, pois a configuração fornecida à aplicação foi de 2 gigabytes de memória RAM e 50 gigabytes de armazenamento, cumprindo os requisitos para funcionamento da aplicação.

O sistema operacional escolhido para gerenciar todo o software precisa de arquitetura de servidores, de cunho livre sem fins lucrativos e utiliza pouco espaço de armazenamento para ser instalado. O Ubuntu Server 22.04 LTS foi escolhido, pois atende todos os requisitos levantados. Sua instalação é intuitiva e pode ser feita mediante a um pendrive de 2 gigabytes. Possui softwares de gerenciamento para que a aplicação seja monitorada. Ocupa cerca de 10 GB de armazenamento após a instalação completa. Não possui interface gráfica, poupando a quantidade de memória RAM.

O banco de dados deve seguir o critério de ser livre e sem fins lucrativos, possuir fácil acesso para o usuário e usar linguagem SQL para sua programação. O banco de dados MariaDB foi o que chegou mais próximo de atender a todos os critérios. Possui fácil instalação no Ubuntu Server e usa linguagem SQL.

Para que a aplicação web desenvolvida pelo Santos (2022a) fosse acessada mediante outros computadores na rede, instalou-se em todas as dependências necessárias usadas no software. Dentre elas, estão as principais: linguagem de programação PHP e o servidor web Apache.

4 Desenvolvimento

Este tópico tem como função demonstrar os procedimentos realizados para a construção do sistema.

4.1 Transmissão

Uma transmissão consiste em enviar informações de uma determinada origem a um determinado destino. Seguindo essa premissa, em uma bancada de teste usou-se o arduino nano em conjunto com o transceptor NR24L01+ para enviar dados aleatórios a um receptor. Esse mesmo transceptor foi montado em outra bancada usando o arduino uno e o mesmo transceptor. Essas bancadas possuíam o intuito de entender como é dado o envio e recebimento dos dados por meio do transceptor em conjunto com o arduino. A figura 7 mostra o sistema de transmissão e recepção montados.

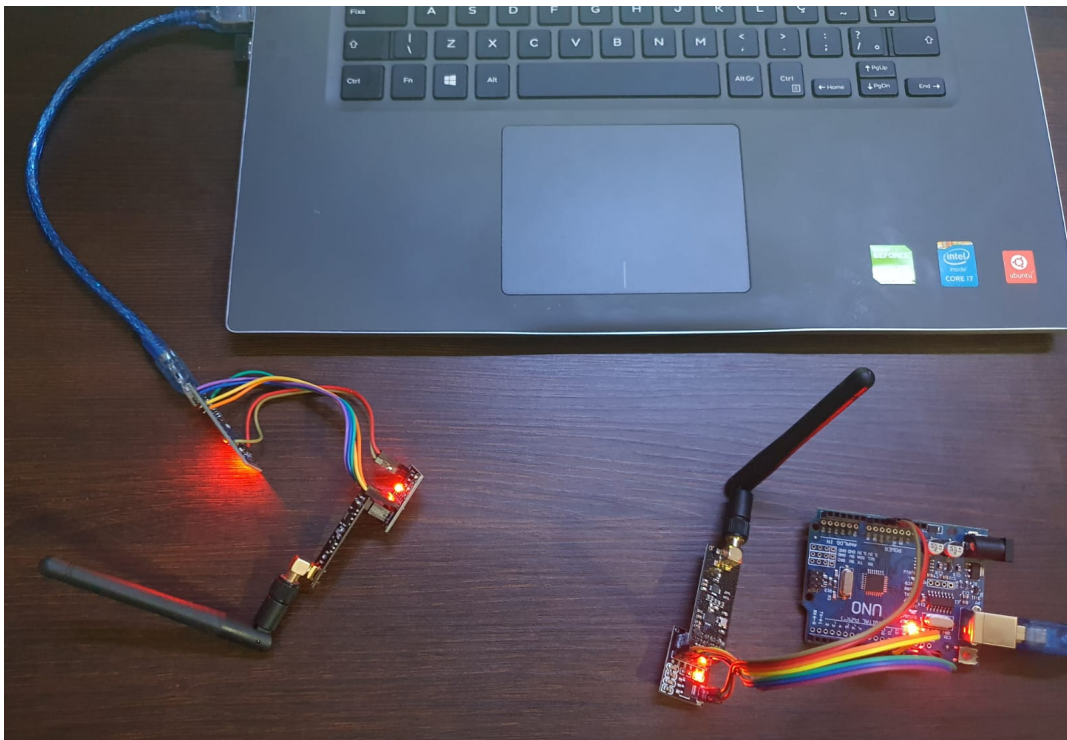


Figura 7 – Bancada de teste para programação do transceptor

A comunicação entre as duas bancas de testes apresentou o resultado esperado mostrado na figura 8 e 9. A figura 8 mostra o código desenvolvido para o transmissor e para o receptor. A figura 9 mostra os dados de testes recebidos pelo receptor na porta COM4.



Figura 8 – Código desenvolvido para a banca de transmissão e recepção

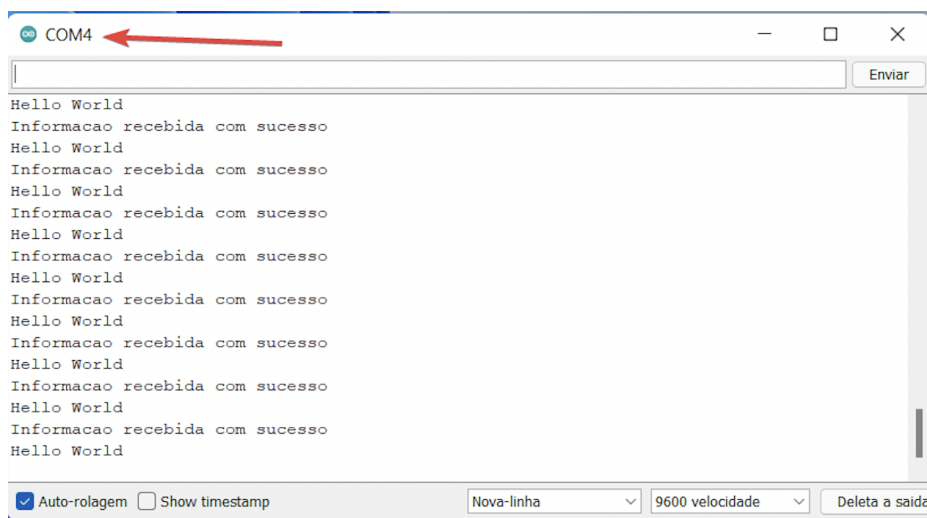


Figura 9 – Dados de teste recebidos pela banca do receptor

4.2 Receptor

Com a parte de recebimento dos dados testada através da bancada de recepção, iniciou-se o entendimento de como seria a comunicação com o banco de dados. Usando o arduino uno em conjunto com o módulo ethernet conectado a rede, instanciou as diretivas de banco de dados por meio da IDE do arduino.

Para o envio das informações ao banco de dados é necessário identificar o endereço de rede do servidor e inseri-lo no código fonte. Esse endereço é definido como IP (Internet Protocol). Os dados de acesso de usuário e senha foram inseridos no código para conceder acesso ao arduino a gravar os dados nas tabelas. As diretivas de banco de dados forma implementadas para inserir os dados segundo o padrão da linguagem SQL. Essa informações são mostradas na figura 10.



```
sketch_aug05a | Arduino 1.8.19
sketch_aug05a
RF24 radio(7, 10); // define as portas do arduino que irão ser usadas para receber os dados

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

IPAddress server_addr(192, 168, 1, 100); // IP of the MySQL *server* here
char user[] = ""; // MySQL user login username
char password[] = ""; // MySQL user login password

char INSERIR_BANCO[] = "INSERT INTO meteorologico.registro (idData,idHora,idEndereco,idEstacao,idUsuario,temperatura,umidadeRelat
char Select1[] = "SELECT idHora FROM meteorologico.hora WHERE hora = '%d:%d:%d'";
char Insert1[] = "INSERT INTO meteorologico.hora (hora) VALUES ('%d:%d:%d')";
char Select2[] = "SELECT idData FROM meteorologico.data WHERE data = '%d-%d-%d'";
char Insert2[] = "INSERT INTO meteorologico.data (data) VALUES ('%d-%d-%d')";
char query[228];

Erro compilando para a placa Arduino Uno Copiar mensagens de erro
0 sketch usa 34376 bytes (106%) de espaço de armazenamento para programas. O máximo são 32256 bytes.
Variáveis globais usam 2303 bytes (112%) de memória dinâmica, deixando -255 bytes para variáveis locais. O máximo são 2048 bytes.
text section exceeds available space in board
Sketch too big; see https://support.arduino.cc/hc/en-us/articles/360013825179 for tips on reducing it.
Erro compilando para a placa Arduino Uno
21 Arduino Uno em /dev/cu.usbserial-A50285BI
```

Figura 10 – Estouro de memória do arduino uno usando as diretivas de banco de dados

No banco de dados desenvolvido por Santos (2022a) possui três tabelas para inserção. Segundo Monk (2015) o Arduino Uno o apresentou pouca memória para lidar com as diretivas de **Select** e **Insert**. Ao carregar as informações para o arduino houve um estouro de memória. A figura 10 mostra que o estouro de memória ultrapassou em 112%. Com esse problema o uso das diretivas principais de inserção e busca de informações do banco de dados do Santos (2022a) se tornou inviável para o Arduino Uno. Para contornar o problema, decidiu-se criar uma nova tabela no banco de dados que receberá as informações da estação meteorológica.

A figura 11 mostra o código fonte da nova tabela que foi desenvolvida para receber os dados da estação.

```

1 CREATE TABLE `registroTemporario` (
2   `idRegistro` int(11) NOT NULL,
3   `Data` int(11) NOT NULL,
4   `Hora` int(11) NOT NULL,
5   `idEndereco` int(11) NOT NULL,
6   `idEstacao` int(11) NOT NULL,
7   `idUsuario` int(11) NOT NULL,
8   `temperatura` decimal(5,2) NOT NULL,
9   `umidadeRelativa` decimal(3,0) NOT NULL,
10  `irradiacaoSolar` decimal(7,0) NOT NULL,
11  `precipitacaoPluviometrica` decimal(6,2) NOT NULL,
12  `velocidadeVento` decimal(4,1) NOT NULL,
13  `direcaoVento` decimal(2,0) NOT NULL
14 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
15
16 ALTER TABLE `registroTemporario`
17 ADD PRIMARY KEY (`idRegistro`),
18 ADD KEY `fk_registro_endereco1_idx` (`idEndereco`),
19 ADD KEY `fk_registro_estacao1_idx` (`idEstacao`),
20 ADD KEY `fk_registro_usuario1_idx` (`idUsuario`);

```

Figura 11 – Nova tabela no banco de dados

4.3 Banco de dados

Para que as tabelas principais recebessem os dados, foi criada uma *trigger*. Uma *trigger* dentro da linguagem SQL é definida como uma ação automática após a tabela vinculada à mesma sofrer alguma uma tentativa de inserção, atualização ou exclusão dos dados, não podendo nunca ser ignorada. Para cada inserção realizada na nova tabela desenvolvida a trigger é acionada copiando os dados da nova tabela para as tabelas principais.

```

1 CREATE TRIGGER inserirTabelasPrincipais
2 ON registroTemporario
3 FOR INSERT
4 AS
5 DECLARE
6 idDate integer;
7 idHour integer;
8 BEGIN
9   SELECT idData from data where data = NEW.data into idDate;
10
11   IF !idDate THEN
12     INSERT INTO data VALUE NEW.date;
13     SET idDate = SELECT idData FROM data where data = NEW.data;
14   END IF;
15
16   SELECT idHora FROM hora where hora = NEW.hora into idHour;
17
18   INSERT INTO registro VALUE (
19     idDate,
20     idHour,
21     NEW.idEstacao,
22     NEW.idUsuariao,
23     NEW.temperatura,
24     NEW.umidadeRelativa,
25     NEW.irradiacaoSolar,
26     NEW.precipitacaoPluviometrica,
27     NEW.velocidadeVento,
28     NEW.direcaoVento
29   );
30
31 END

```

Figura 12 – Trigger desenvolvida para inserir os dados nas tabelas principais

Na figura 12 é apresentado o código da *trigger* desenvolvido para contornar o problema do estouro de memória do arduino e alimentação dos dados nas tabelas corretas. A cada nova inserção realizada na tabela **registroTemporario** a *trigger* é acionada. Na linha 9 da figura 12, é feito uma busca na tabela **data** e verifica se a data enviada pela estação já está armazenada. Caso não esteja, entre as linhas 11 e 14 há um condicional que insere a nova data na tabela correta e armazena o código da nova data na variável **idDate**.

Na linha 16 da figura 12 é feita uma busca na tabela hora para encontrar o código correspondente à hora informada pela estação. Esse código é armazenado na variável **idHour**. Na linha 18 à 29, todos os dados recebidos são inseridos na tabela registro.

4.4 Sistema geral

Para entendimento do funcionamento de todos os componentes em conjunto foi usado o software Cisco Packet Tracer para simular os componentes trabalhando em grupo.

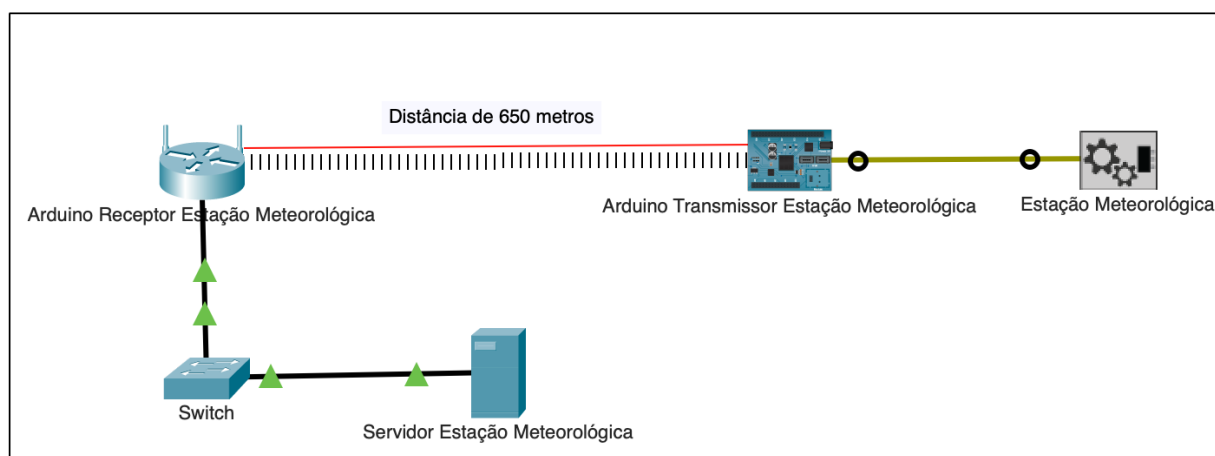


Figura 13 – Esquema do sistema de comunicação da estação meteorológica com o servidor.

A figura 13 mostra o funcionamento de todos os componentes juntos. Os dados meteorológicos são coletados pela estação e enviados ao **Arduino Transmissor** via porta digital. O transmissor envia um pacote de dados de cada sensor da estação por vez. O receptor coleta os dados, armazenando-os em variáveis internas. As variáveis são passadas como parâmetro para a função de inserção do banco de dados. O receptor envia os dados ao servidor por meio do protocolo TCP/IP.

Por meio do protocolo TCP/IP, o **Arduino Receptor** encontra o IP do **Servidor Estação Meteorológica** na rede e o TCP garante a entrega das informações. Gravadas as informações na tabela temporária no bando de dados, a trigger é acionada e realiza as inserções nas tabelas principais.

5 Resultados e Discussão

O sistema foi implementado no Instituto Federal Goiano – Campus Rio Verde, Laboratório de Automação, Simulação e Controle, onde também foi feita a montagem dos módulos de transmissão, recepção e teste, processo descrito nos tópicos seguintes.

5.1 Alteração da estação

O Arduino Nano e o módulo transceptor NRF24L01+, usados para transmitir os dados da estação meteorológica através de radiofrequência, foram acondicionados em uma caixa patola e instalados dentro do gabinete da estação meteorológica, conforme mostrado na figura 14.



Figura 14 – Instalação do módulo transmissor

A conexão do dispositivo transmissor com a estação meteorológica foi feita através

de um conector circular de 5 pinos, para alimentação de 5V e a comunicação serial síncrona. A estação meteorológica é controlada pelo PIC18F4680, apresentado na figura 15.

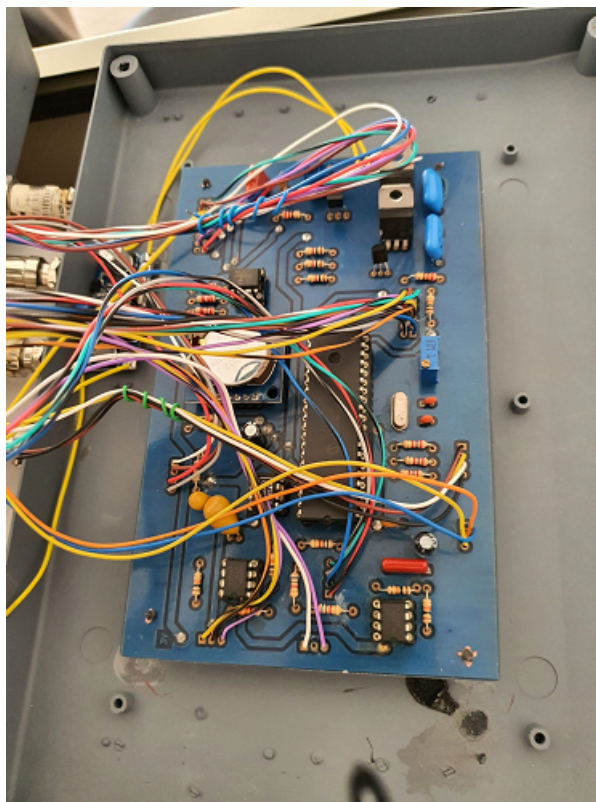


Figura 15 – Controlador PIC18F480 da estação meteorológica

Para o envio dos dados, o programa do microcontrolador pic, desenvolvido em C, através do compilador mikroc Pro for Pic, foi alterado para a implementação do algoritmo de comunicação descrito no Algoritmo 1. Os testes de envio foram inicialmente realizados como um módulo do PIC, mostrado na figura 16, para validar o funcionamento da comunicação serial e verificar o funcionamento do transmissor.

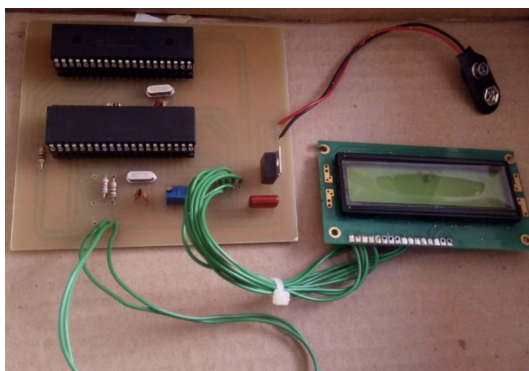


Figura 16 – Circuito de teste para simulação dos dados da estação

A antena transmissora foi instalada na parte superior do gabinete da estação meteorológica e o transmissor foi configurado para trabalhar com a potência máxima no

momento da transmissão. Durante o tempo de espera o módulo foi colocado no modo sleep para economia de energia. O código fonte gerado é mostrado no anexo A.



Figura 17 – Local do teste de alcance do transceptor

Um teste do transceptor foi feito para medir o alcance em metros do envio das informações. A figura 17 mostra o local onde o teste foi realizado. O ponto inicial representado pelo número 0 foi o local onde o transmissor foi deixado. O teste baseou-se em caminhar com o receptor em linha reta até o momento em que ele não recebe mais as informações. O resultado foi um alcance de 800 metros.

Como o sinal de radiofrequência pode sofrer interferência não se pode trabalhar em sua distância máximo. A distância de 650 metros foi determinado como margem de segurança para prevenir possíveis interferências ao transmissor e ao receptor.

5.2 Receptor

Na figura 18, apresenta-se a fabricação da placa desenvolvida no software Kicad. O módulo transceptor NR24L01+ e os botões de pulso, usados para receber os dados do transmissor e manipulação da interface do usuário, foram acondicionados na placa de circuito impresso, mostrado na figura 19.

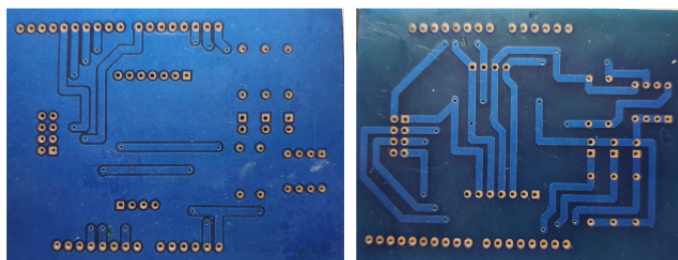


Figura 18 – Placa de circuito impresso fabricada

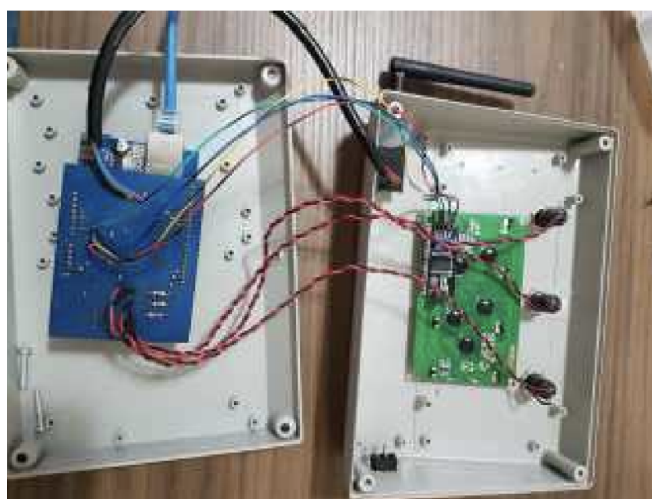


Figura 19 – Receptor montado

A conexão do dispositivo receptor com a rede de internet se deu através do cabo de internet ligado direto ao módulo Ethernet. A alimentação do receptor é feita por uma fonte externa de 12V, ligada ao Arduino Uno. Na figura 19, o Arduino Uno, o módulo Ethernet e a placa de circuito impresso foram acondicionados dentro de uma caixa de patola.

Devido ao problema de falta de memória do Arduino Uno para lidar com as diretivas de banco de dados SQL, a lógica de programação foi alterada para realizar a comunicação com a nova tabela desenvolvida. O anexo B mostra todo o código desenvolvido em torno da nova tabela e recebimentos dos dados do transmissor. A comunicação entre o receptor e o servidor mostrou-se eficiente, enviando todos os dados da estação corretamente ao banco de dados.

5.3 Banco de dados

A nova tabela proposta para contornar o problema de pouca memória do Arduino Uno mostrou-se satisfatória para armazenar os dados da estação meteorológica. A trigger desenvolvida mostrou-se eficiente. A cada inserção na nova tabela, a trigger é acionada e realiza todas as inserções nas tabelas principais como o esperado.

5.4 Servidor

O Ubuntu Server 22.04 LTS foi instalado no servidor. Apresentou perfeita compatibilidade com a interface web inserida. Possui 2 gigabytes de memória RAM disponível e 50 gigabytes de armazenamento para gerenciamento. Para funcionamento do software web em PHP, foi instaladas as dependências fornecidas por Santos (2022a).

O IP gerado para o servidor foi fixado dentro dos diretórios do site. Foi alterado o nome da pasta **AJAX** para **ajax** para atender as requisições de diretório. Permissão de leitura e escrita foram concedidas aos diretórios para a gravação dos arquivos que a interface gera.

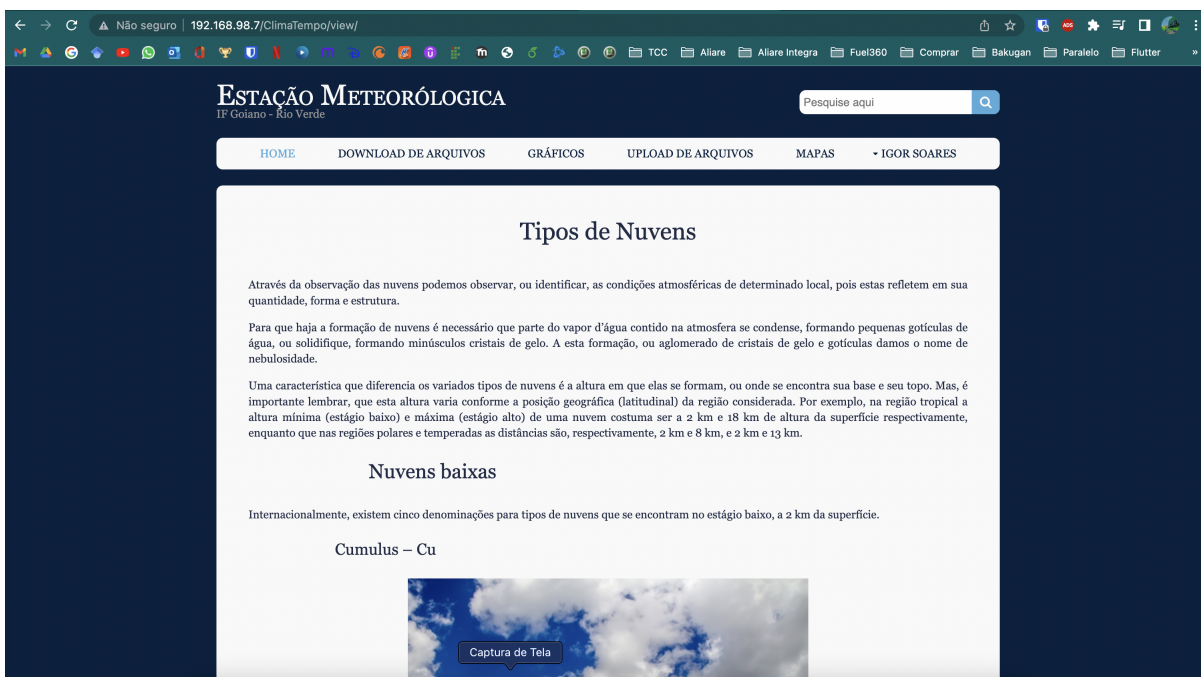


Figura 20 – Site hospedado no servidor do IFGoiano - Campus Rio Verde

O IP estabelecido para o software web foi o 192.168.98.7. Na figura 14, mostra o funcionamento da interface. Todas as telas e ações inseridas funcionam como o esperado.

6 Conclusão

Esse trabalho foi realizado para apresentar a alteração de uma estação meteorológica semiautomática para uma estação meteorológica automática, a partir da utilização de microcontroladores e transceptores como ponte de comunicação entre a estação e o banco de dados. As estações automáticas tem ganhado força em várias áreas que procuram aliar os dados climáticos com o manejo da agricultura, fornecendo dados para o cálculo da evapotranspiração.

Os objetivos propostos para esse trabalho inicialmente, eram capacitar a estação meteorológica, situada no Laboratório de Automação, Simulação e Controle, a enviar dados sem ajuda humana por meio de um transmissor. Um receptor seria desenvolvido para receber os dados da estação e enviá-los ao banco de dados. A interface web desenvolvida por Santos (2022a) seria hospedada em um servidor, para análise dos dados armazenados e interação do usuário.

Pelo que foi exposto e os resultados obtidos durante o desenvolvimento e testes, conclui-se que as modificações atenderam de maneira satisfatória aos objetivos propostos, pois é capaz de realizar todas as ações previstas nos requisitos do sistema.

Apesar de ter resultados satisfatório, a estação apresenta limitações, pois os testes indicaram que possui um alcance de 800 m sem obstáculos e pode sofrer interferência externa de outros equipamentos, que emitem ondas de rádio.

Várias dificuldades foram encontradas durante o desenvolvimento deste trabalho, sendo que a principal delas foi a configuração do servidor web, para ser utilizado pelo usuário e a falta de memória do arduino uno, para lidar com as diretivas de banco de dados em seu código fonte. A maioria dos microcontroladores que usam o chip ATmega328 possuem essa limitação de memória para armazenar o código fonte e expansão das funcionalidades, o que não permitiu atingir os requisitos de inserção do banco de dados conforme os requisitos.

O sistema permite melhorias, as quais pode ser realizadas em trabalhos futuros, sendo que algumas delas são apresentadas a seguir.

Mesmo tendo utilizado microcontroladores com pouca memória e transceptores de curto alcance, a parte lógica e física possibilita uma grande ampliação. Pode incluir novas antenas que consigam enviar dados em longas distâncias e que sofram pouca interferência. Alterar o microcontrolador do receptor para maior manipulação dos dados sem grandes limitações de memória.

Outra sugestão, para o futuro, seria a implementação do protocolo Wifi no receptor, para enviar os dados sem uso da rede cabeada de internet. Também, desenvolvimento de uma nova interface que possibilite análise dos dados em tempo real e via aplicativo mobile.

Por fim, o desenvolvimento deste trabalho deve auxiliar estudantes e professores na área da automação e agricultura, pois possui a implementação de vários conceitos

importantes para ambas as áreas e pode servir como base de desenvolvimento para trabalhos futuros.

Referências

- ARDUINO. *Arduino Nano*. 2022. Disponível em: <<https://store-usa.arduino.cc/products/arduino-nano?selectedStore=us>>. Citado na página 10.
- ARDUINO. *O que é Arduino?* 2022. Disponível em: <<https://www.arduino.cc/en/Guide/Introduction>>. Citado 4 vezes nas páginas , 9, 10 e 13.
- ENGINEERS, L. M. *How nRF24L01+ Wireless Module Works Interface with Arduino*. 2022. Disponível em: <<https://lastminuteengineers.com/nrf24l01-arduino-wireless-communication/>>. Citado 3 vezes nas páginas 10, 11 e 14.
- FINHOLDT, G. et al. Desenvolvimento de uma estação meteorológica automática para manejo de irrigação. *Revista Engenharia na Agricultura-REVENG*, v. 19, n. 4, p. 313–328, 2011. Citado 2 vezes nas páginas 1 e 2.
- HAYKIN, S.; MOHER, M. *Sistemas Modernos de Comunicação Wireless*. [S.l.]: Bookman, 2008. v. 1. Citado 2 vezes nas páginas 7 e 10.
- INMET. 2022. Disponível em: <<https://portal.inmet.gov.br/glossario/glossario#M>>. Citado 6 vezes nas páginas , 1, 3, 4, 5 e 8.
- IOANNOU, K. et al. Low-cost automatic weather stations in the internet of things. *Information*, MDPI, v. 12, 2021. Citado na página 6.
- LEITE, C. V. et al. Reference evapotranspiration through hargreaves method using the solar radiation estimation for goiás state, brazil. *Revista Engenharia na Agricultura*, v. 28, p. 274–292, 2020. Citado na página 8.
- MILANEZ, A. Y. et al. Conectividade rural: situação atual e alternativas para superação da principal barreira à agricultura 4.0 no brasil. Banco Nacional de Desenvolvimento Econômico e Social, 2020. Citado na página 8.
- MONK, S. *Programação com Arduino II: Passos avançados com sketches*. [S.l.]: Bookman Editora, 2015. Citado na página 19.
- NASUTI, S. et al. Conhecimento tradicional e previsões meteorológicas: Agricultores familiares e as “experiências de inverno” no semiárido potiguar. *Revista de Economia do Nordeste*, v. 44, p. 383–402, 2013. Citado na página 3.
- OLIVEIRA, L. A. N.; ALENCAR, M. S.; LOPES, W. T. A. Evolução da arquitetura de redes móveis rumo ao 5g. 2018. Citado 2 vezes nas páginas 7 e 8.
- ORGANIZATION, W. M. *Guide to Instruments and Methods of Observation; Volume III—Observing Systems*. 2021. Disponível em: <https://community.wmo.int/activity-areas/imop/wmo-no_8>. Acesso em: 23 outubro 2021. Citado na página 5.
- PERAZZI, P. R. et al. O tradicional ou o moderno? uma visão da informação da rede de estações meteorológicas brasileiras. *Revista Brasileira de Meteorologia*, Rbmet, v. 36, n. 3, p. 146–366, 2021. Citado 2 vezes nas páginas 4 e 5.

PINHEIRO, J. *Guia completo de cabeamento de redes*. [S.l.]: Elsevier Brasil, 2016. Citado na página 6.

RIBEIRO, A. A.; SIMEÃO, M.; SANTOS, A. R. B. Comparação de métodos de estimativa da evapotranspiração de referência no período chuvoso e seco em piripiri (pi). *Revista Agrogeoambiental*, v. 8, p. 89–100, 2016. Citado na página 3.

SANTOS, I. S. D. *DESENVOLVIMENTO DE APLICAÇÕES WEB PARA MANIPULAÇÃO E CONTROLE DE DADOS METEOROLÓGICOS*. 2022. Disponível em: <<https://www.even3.com.br/anais/ceictifgoiano/60064-desenvolvimento-de-aplicacoes-web-para-manipulacao-e-controle-de-dados-meteorologicos>>. Citado 4 vezes nas páginas 16, 19, 26 e 27.

SANTOS, I. S. D. *DESENVOLVIMENTO DE UMA ESTAÇÃO METEOROLÓGICA MÓVEL APLICAÇÕES NO AMBIENTE AGRÍCOLA*. 2022. Disponível em: <<https://www.even3.com.br/anais/ceictifgoiano/60580-desenvolvimento-de-uma-estacao-meteorologica-movel-aplicacoes-no-ambiente-agricola/>>. Citado 2 vezes nas páginas 10 e 11.

SEMICONDUCTOR, N. nrf24l01+ single chip 2.4 ghz transceiver product specification v1. 0. *Nordic Semiconductors, Norway*, 2008. Citado na página 13.

SILVA, D. d.; VALE, F. d. Tomate: tecnologia de produção. *Editora UFV, Viçosa, Brazil*, 2007. Citado na página 1.

SILVA, E. B. da et al. *Automação & Sociedade: Quarta Revolução Industrial, um olhar para o Brasil*. [S.l.]: Brasport, 2018. Citado na página 1.

SOUZA, B. Estudo de normais climatológicas para estações meteorológicas do tipo convencional. *Universidade Federal de Santa Catarina. Centro de Filosofia e Ciências Humanas. Geografia*, 2018. Citado na página 5.

TANENBAUM, A. S. *Redes de Computadores*. [S.l.]: Pearson Education do Brasil Ltda, 2011. v. 5. Citado 2 vezes nas páginas 7 e 10.

YNOUE, R. Y. et al. *Meteorologia noções básicas*. [S.l.]: Oficina de texto, 2017. v. 1. Citado na página 3.

Anexos

ANEXO A – Código fonte do transmissor

```
1 #include <SPI.h>
2 #include <nRF24L01.h>
3 #include <RF24.h>
4
5 RF24 radio(9, 8);
6
7 const byte address[6] = "00001";
8 int temp = 0;
9 int umid = 0;
10 int rad = 0;
11 int pluv = 0;
12 int vel = 0;
13
14 int porta6 = 6;
15 int porta7 = 7;
16 void setup()
17 {
18   radio.begin();
19
20   pinMode(porta6, INPUT);
21   pinMode(porta7, INPUT);
22
23   radio.openWritingPipe(address);
24   radio.setPALevel(RF24_PA_HIGH);
25   radio.stopListening();
26 }
27
28 void loop()
29 {
30   while (digitalRead(porta6) == HIGH && digitalRead(porta7) == HIGH);
31   ler_dados();
32
33 }
34
35 void ler_dados()
36 {
37   unsigned char dados[80];
38   int n, k;
39   k = 0;
40
41   dados[0] = digitalRead(porta7);
42   k++;
43   while (digitalRead(porta6) == HIGH)
```

```
44     ;
45     for (n = 1; n < 80; n++)
46     {
47         while (digitalRead(porta6) == HIGH && digitalRead(porta7) == HIGH)
48             ;
49
50         if (digitalRead(porta7) == LOW)
51             break;
52
53         dados[n] = digitalRead(porta7);
54
55         k++;
56         while (digitalRead(porta6) == HIGH);
57     }
58     delay(300);
59     temp = 0;
60     if (k == 80)
61     {
62         for (n = 15; n >= 0; n--)
63             temp = temp << 1 + dados[n];
64
65         umid = 0;
66         for (n = 15; n >= 0; n--)
67             umid = umid << 1 + dados[n + 16];
68
69         vel = 0;
70         for (n = 15; n >= 0; n--)
71             vel = vel << 1 + dados[n + 32];
72
73         pluv = 0;
74         for (n = 15; n >= 0; n--)
75             pluv = pluv << 1 + dados[n + 48];
76
77         rad = 0;
78         for (n = 15; n >= 0; n--)
79             rad = rad << 1 + dados[n + 64];
80     }
81     else
82     {
83         console.log("Falha na leitura dos dados da esta o.")
84     }
85
86     if (radio.available())
87     {
88         radio.write(&temp, sizeof(temp));
89         Serial.println(temp);
90
```

```
91     radio.write(&umid, sizeof(umid));
92     Serial.println(umid);
93
94     radio.write(&rad, sizeof(rad));
95     Serial.println(rad);
96
97     radio.write(&pluv, sizeof(pluv));
98     Serial.println(pluv);
99
100    radio.write(&velo, sizeof(vel));
101    Serial.println(vel);
102
103    if (temp != temp1)
104        temp1 = temp;
105    if (umi != umi1)
106        umi1 = umi;
107    if (rad != rad1)
108        rad1 = rad;
109    if (prec != prec1)
110        prec1 = prec;
111    if (velo != velo1)
112        velo1 = velo;
113 }
114 }
```

Listing A.1 – Código fonte do transmissor

ANEXO B – Código fonte do receptor

```

1 #include <Ethernet.h>
2 #include <MySQL_Connection.h>
3 #include <MySQL_Cursor.h>
4 #include <Wire.h>
5 #include <LCD.h>
6 #include <LiquidCrystal_I2C.h>
7 #include <SPI.h>
8 #include <RF24L01.h>
9 #include <RF24.h>
10 #include <DS1307RTC.h>
11
12 byte mac_addr[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
13 const byte address[6] = "00001";
14
15 RF24 radio(7, 10);
16
17 LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
18
19 IPAddress server_addr(192, 168, 98, 7);
20 char user[] = "";
21 char password[] = "";
22
23 char INSERIR_BANCO[] = "INSERT INTO meteorologico.registro (Data,Hora,
      idEndereco ,idEstacao ,idUserio ,temperatura ,umidadeRelativa ,
      irradiacaoSolar ,precipitacaoPluviometrica ,velocidadeVento ,
      direcaoVento) VALUES (%d,%d,%d,%d,%d,%lf,%lf,%lf,%lf,%lf,%s)";
24
25 char query[228];
26
27
28 double temp = 0;
29 double umi = 0;
30 double rad = 0;
31 double prec = 0;
32 double velo = 0;
33 char direc;
34 double temp1 = 0;
35 double umi1 = 0;
36 double rad1 = 0;
37 double prec1 = 0;
38 double velo1 = 0;
39 char direc1;
40

```

```
41
42 int G = 0;
43 int g = 0;
44 int Botao1 = 3;
45 int Botao2 = 2;
46 int Botao3 = 4;
47
48 EthernetClient client;
49 MySQL_Connection conn((Client *)&client);
50
51 void setup()
52 {
53
54   Serial.begin(115200);
55
56   lcd.begin(20, 4);
57   radio.begin();
58
59   pinMode(Botao1, INPUT);
60   pinMode(Botao2, INPUT);
61   pinMode(Botao3, INPUT);
62
63   radio.openReadingPipe(0, address);
64   radio.startListening();
65
66   Ethernet.begin(mac_addr);
67
68   lcd.setCursor(0, 0);
69   lcd.print("Iniciando Sistema");
70   delay(3000);
71   if (conn.connect(server_addr, 3306, user, password))
72   {
73     lcd.clear();
74     lcd.setCursor(0,0);
75     lcd.print("Conectado ao BD");
76     delay(5000);
77   }
78   else
79   {
80     lcd.clear();
81     lcd.setCursor(0, 0);
82     lcd.print("Connection failed with BD");
83     Serial.println("Connection failed...:(");
84     delay(5000);
85   }
86
87   lcd.clear();
```

```
88
89 }
90
91 void loop() {
92
93   if (digitalRead(Botao1) == HIGH && digitalRead(Botao2) == HIGH) {
94     Menu();
95   }
96
97   if (digitalRead(Botao3) == HIGH) {
98     Configuracao();
99   }
100
101   lcd.setCursor(0,0);
102   lcd.print("Modulo de receptcao");
103
104   if(g == 1){
105     if (radio.available())
106     {
107       radio.read(&hora, sizeof(hora));
108       Serial.println(hora);
109
110       radio.read(&data, sizeof(data));
111       Serial.println(data);
112
113       radio.read(&temp, sizeof(temp));
114       Serial.println(temp);
115
116       radio.read(&umi, sizeof(umi));
117       Serial.println(umi);
118
119       radio.read(&rad, sizeof(rad));
120       Serial.println(rad);
121
122       radio.read(&prec, sizeof(prec));
123       Serial.println(prec);
124
125       radio.read(&velo, sizeof(velo));
126       Serial.println(velo);
127
128       radio.read(&direc, sizeof(direc));
129       Serial.println(direc);
130
131       if(hora != hora1) hora1 = hora;
132       if(minu != minu1) minu1 = minu;
133       if(seg != seg1) seg1 = seg;
134       if(dia != dia1) dia1 = dia;
```



```
135     if(mes != mes1) mes1 = mes;
136     if(ano != ano1) ano1 = ano;
137     if(temp != temp1) temp1 = temp;
138     if(umi != umi1) umi1 = umi;
139     if(rad != rad1) rad1 = rad;
140     if(prec != prec1) prec1 = prec;
141     if(velo != velo1) velo1 = velo;
142     if(direc != direc1) direc1 = direc;
143
144
145
146     lcd.setCursor(0,0);
147     lcd.print(G);
148     lcd.setCursor(0,2)
149     lcd.print("Gravacao");
150
151     Banco_de_Dados();
152 }
153
154 }
155
156 }
157
158
159 void Banco_de_Dados(){
160
161     MySQL_Cursor *cur_mem = new MySQL_Cursor(&conn);
162
163     sprintf(query, INSERIR_BANCO,data,hora,1,152,1,temp1, umi1, rad1,
164             prec1, velo1, direc1);
165
166     cur_mem->execute(query);
167
168     delete cur_mem;
169
170     G = G+1;
171 }
172
173 void Configuracao(){
174     if(g == 0){
175
176         lcd.clear();
177         lcd.setCursor(0, 0);
178         lcd.print("Iniciar Gravacao?");
179         lcd.setCursor(0,1);
180         lcd.print("(S/N)");
```

```
181     delay(800);
182     while(digitalRead(Botao1)== LOW && digitalRead(Botao2)==LOW);
183     delay(300);
184
185
186     if(digitalRead(Botao1) == HIGH){
187         lcd.clear();
188         g = 1;
189         G = 0;
190         return;
191     }
192 } else if (g == 1){
193
194     lcd.clear();
195     lcd.setCursor(0, 0);
196     lcd.print("Parar Gravacao?");
197     lcd.setCursor(0,2);
198     lcd.print("(S/N)");
199     delay(800);
200     while(digitalRead(Botao1)== LOW && digitalRead(Botao2)==LOW);
201     delay(200);
202
203     if(digitalRead(Botao1) == HIGH){
204         lcd.clear();
205         g = 0;
206         G = 0;
207         return;
208     }
209 }
210 lcd.clear();
211 }
```

Listing B.1 – Código fonte receptor