

UNIVERSIDADE FEDERAL DO PARANÁ

RODRIGO KOGERATSKI

**SISTEMA DE MONITORAMENTO PARA ESTUFAS DE CULTIVO DE AGARICUS
BISPORUS**

CURITIBA

2019

RODRIGO KOGERATSKI

**SISTEMA DE MONITORAMENTO PARA ESTUFAS DE CULTIVO DE AGARICUS
BISPORUS**

Trabalho apresentado à disciplina TE298
Trabalho de Conclusão de Curso B como
requisito parcial à conclusão do Curso de
Engenharia Elétrica – Ênfase em
Sistemas Eletrônicos Embarcados, Setor
de Tecnologia, da Universidade Federal
do Paraná.

Orientador: Prof. Dr. James Alexandre
Baraniuk

CURITIBA

2019

TERMO DE APROVAÇÃO

RODRIGO KOGERATSKI

SISTEMA DE MONITORAMENTO PARA ESTUFAS DE CULTIVO DE AGARICUS BISPORUS

Trabalho apresentado à disciplina TE298 Trabalho de Conclusão de Curso B como requisito parcial à conclusão do Curso de Engenharia Elétrica – Ênfase em Sistemas Eletrônicos Embarcados, Setor Setor de Tecnologia, da Universidade Federal do Paraná.

Orientador: Prof. Dr. James Alexandre Baraniuk
Departamento de Engenharia Elétrica, UFPR

Prof. Dr. André Bellin Mariano
Departamento de Engenharia Elétrica, UFPR

Prof. Ph.D. Patricio Rodolfo Impinnisi
Departamento de Engenharia Elétrica, UFPR

Curitiba, 22 de novembro de 2019.

SUMÁRIO

1. INTRODUÇÃO	6
1.1. OBJETIVOS	6
1.1.1. OBJETIVO GERAL	6
1.1.2. OBJETIVOS ESPECÍFICOS	7
1.2. JUSTIFICATIVA	7
2. FUNDAMENTAÇÃO TEÓRICA	9
2.1. PROCESSO DE CULTIVO DO AGARICUS BISPORUS	9
2.2. MEDIÇÃO DE TEMPERATURA	11
2.2.1. TERMISTORES	11
2.2.2. TERMOPARES	12
2.2.3. TERMORESISTORES	12
2.2.4. PAR BIMETÁLICO	13
2.2.5. SENSORES ELETRÔNICOS	14
2.2.6. PIRÔMETRO	14
2.3. MEDIÇÃO DE UMIDADE RELATIVA DO AR	15
2.4. MEDIÇÃO DE CORRENTE ELÉTRICA	16
2.4.1. MEDIÇÃO DE CORRENTE CONTÍNUA POR RESISTOR SHUNT	16
2.4.2. MEDIÇÃO DE CORRENTE CONTÍNUA POR EFEITO HALL	17
2.4.3. MEDIÇÃO DE CORRENTE ALTERNADA POR MEIO DE TC	18
2.5. GRAU DE PROTEÇÃO IP – NORMA IEC60529	19
2.6. BANCO DE DADOS	20
2.7. LINGUAGEM DE PROGRAMAÇÃO PYTHON	22
2.8. LINGUAGEM DE PROGRAMAÇÃO JAVA SCRIPT	23
2.9. REDES DE COMPUTADORES	24
2.10. PROTOCOLO ETHERNET	24
3. DESENVOLVIMENTO	26
3.1. PROPOSTA GERAL DO PROJETO	26
3.2. AVALIAÇÃO DO PROCESSO DE CULTIVO	27
3.3. SELEÇÃO DOS INSTRUMENTOS E EQUIPAMENTOS	28
3.3.1. SINGLE BOARD COMPUTER - SBC	28
3.3.2. PAINEL DE AUTOMAÇÃO	30
3.3.3. SENSOR DE TEMPERATURA E UMIDADE	31
3.3.4. MEDIDOR DE CONSUMO DE ENERGIA	34
3.3.5. SENSOR DE CO ₂	35

3.4. CONTROLE DA TEMPERATURA	37
3.5. CONTROLE DA UMIDADE	39
3.6. CONTROLE DO NÍVEL DE CO ₂ E SISTEMA DE VENTILAÇÃO	40
3.7. SERVIDOR E SISTEMA SUPERVISÓRIO	40
3.8. SOFTWARE PRINCIPAL	43
3.9. BANCO DE DADOS	43
3.9.1. BANCO DE DADOS RELACIONAL VS NÃO RELACIONAL	43
3.9.2. TOPOLOGIA DO BANCO DE DADOS	45
3.9.3. BANCO DE DADOS EMBARCADO SQLITE	45
3.10. USO DE CONTÊINERES COMO ESTUFAS DE CULTIVO	46
3.11. ANÁLISE DE VIABILIDADE ECONÔMICA	49
3.11.1. HIPÓTESE ESTUFA COM POUCO RECURSO TECNOLÓGICO	49
3.11.2. HIPÓTESE ESTUFA COM RECURSOS TECNOLÓGICOS	50
3.11.3. INVESTIMENTO E PAYBACK	51
3.12. PROJETO PAINEL DE AUTOMAÇÃO	53
3.13. INFRAESTRUTURA DE MONTAGEM	54
3.14. ESTUDO DE CASO – ESTUFA DE CULTIVO	56
4. RESULTADOS E DISCUSSÃO	62
4.1. OBSERVAÇÕES, PROBLEMAS E DIFICULDADES	63
4.2. OPORTUNIDADES DE MELHORIAS	66
5. CONSIDERAÇÕES FINAIS E ESTUDOS FUTUROS	67
5.1. ESTUDOS FUTUROS	67
6. REFERÊNCIAS BIBLIOGRÁFICAS	68
APÊNDICE A - CÓDIGO DO SOFTWARE MAIN.PY	71
APÊNDICE B - CÓDIGO DO SERVIDOR NODE_SERVER.JS	76
APÊNDICE C - CÓDIGO DA PÁGINA HTML INDEX.HTML	77
APÊNDICE D - ALGORITMO E <i>MEMBERSHIP FUNCTIONS</i>	94
APÊNDICE E - SIMULAÇÕES DA POTÊNCIA DE SAÍDA DO CONTROLADOR	99
APÊNDICE F - CÁLCULO DA PERDA DE CALOR CONTÊINER PARA O AMBIENTE	102
APÊNDICE G - PROJETO PAINEL DE AUTOMAÇÃO	104

1. INTRODUÇÃO

O cultivo de cogumelos é um processo caro e bastante complexo. De acordo com uma pesquisa feita por Vargas (2011), são algumas das dificuldades enfrentadas por pequenos produtores do Distrito Federal: alto custo de produção, constante controle da umidade relativa do ar, cultura muito exigente e alto envolvimento por parte do produtor.

Contabilizar todos os gastos com insumos da cadeia produtiva, é uma forma de controlar a produção e reduzir o custo dela. Com base no trabalho desenvolvido por Araujo et al. (2016, p. 113), o autor chegou a conclusão que mesmo os pequenos produtores rurais, buscam otimizar seu processo produtivo buscando uma maior eficiência. Segundo ele a contabilidade de custos é uma ferramenta importante para a produção de cogumelos champignon, e em seu estudo de caso, foi possível comprovar a lucratividade do produtor, que colaborou com o estudo, demonstrando que os dados estatísticos levantados puderam ser usados como base para tomadas de decisão.

Este projeto objetiva ser uma ferramenta de apoio para incentivar os produtores de cogumelos do tipo *Agaricus bisporus* (Cogumelo de Paris, também conhecido como *Champignon*), tornando o processo de cultivo mais fácil, principalmente para os iniciantes no ramo, pois uma vez que se conhece o processo, com o sistema de monitoramento pode-se determinar com facilidade se as variáveis medidas que influenciam o processo estão dentro dos limites estipulados.

A automação é uma ferramenta que pode otimizar o processo de cultivo desse tipo de alimento. “Os processos artesanais que antes caracterizavam a produção foram substituídos por sistemas automatizados e controlados eletronicamente” (TOMAZELA, 2017). Apesar da automação ser um sinônimo de alto custo, atualmente com as tecnologias de SBCs é possível criar projetos de desenvolvimento com um custo relativamente baixo e tempo reduzido.

1.1. OBJETIVOS

1.1.1. OBJETIVO GERAL

Projetar uma solução para monitoramento e controle de estufas de cultivo de cogumelos *Agaricus bisporus*, usando a tecnologia de “Single Board Computers – SBCs”.

1.1.2. OBJETIVOS ESPECÍFICOS

Os objetivos específicos do trabalho são:

- i. Caracterizar o processo de cultivo deste gênero de cogumelos, levantando quais são as variáveis de processo que necessitam de monitoramento e controle;
- ii. Programar um servidor (comunicação Ethernet servidor/cliente) com recursos de roteamento, para hospedar as páginas HTML, com as informações mais relevantes do processo de cultivo;
- iii. Utilizar recursos do Javascript para dar uma maior dinâmica para as páginas HTML, programando gráficos com recursos interativos (pan + zoom), e também o recurso Ajax, para atualizar as informações apresentadas, sem a necessidade de atualizar as páginas HTML;
- iv. Fazer uma análise comparativa para determinar qual tipo de banco de dados será o mais adequado para armazenar as informações e quais os equipamentos/instrumentos mais adequados para o cultivo de cogumelos;
- v. Validar esse projeto com um estudo de caso.

1.2. JUSTIFICATIVA

As pessoas estão cada vez mais preocupadas em ter uma alimentação mais saudável. “Os brasileiros estão procurando se alimentar melhor, e isto significa maior consumo de produtos frescos e orgânicos” (MILHORANCE, 2014). Uma das formas que as pessoas estão usando para atingir esse objetivo é aderindo ao vegetarianismo e veganismo, segundo Kanda e Rozário (2017) o crescimento de pessoas que são adeptas a causa vegetariana tem feito com que a indústria do ramo aprimore seus produtos, atendendo as pessoas que possuem este hábito alimentar e também pessoas que buscam uma forma de viver mais saudável e sustentável. Essa parcela da população está usando os cogumelos como fonte de proteínas alternativa, que antes eram provenientes de carnes, “Você sabia que o cogumelo pode substituir um bife? Isso porque este fungo tem alto valor proteico, [...]” (REVISTA SOCIEDADE DA MESA, 2017).

De acordo com Eira, (2012) o alto custo pago pelos cogumelos é um problema para a sociedade e afeta o lucro dos produtores. Para ele este problema pode ser combatido com tecnologias de cultivo, fazendo com que o alimento se adapte melhor às condições do país e desta forma torne-se mais acessível a

população. A produção de cogumelos também se mostra um negócio sustentável causando um impacto muito pequeno no meio ambiente. Segundo Teixeira, (2017), citado por Portela, (2017) "O melhor de tudo é que esse é um negócio limpo em todos os sentidos: ambiental, ético e socialmente falando."

A produção de cogumelos também pode ser uma fonte de renda para famílias que vivem em áreas rurais e que possuem pequenas propriedades de terra. "O cultivo de cogumelos no Brasil apresenta-se como uma alternativa viável para pequenos a médios produtores, [...]" (EIRA, 2012).

2. FUNDAMENTAÇÃO TEÓRICA

As pessoas estão cada vez mais preocupadas em ter uma alimentação mais saudável. “Os brasileiros estão procurando se alimentar melhor, e isto significa maior consumo de produtos frescos e orgânicos” (MILHORANCE, 2014). Uma das formas que as pessoas estão usando para atingir esse objetivo é aderindo ao vegetarianismo e veganismo, segundo Kanda e Rozário (2017) o crescimento de pessoas que são adeptas a causa vegetariana tem feito com que a indústria do ramo aprimore seus produtos, atendendo as pessoas que possuem este hábito alimentar e também pessoas que buscam uma forma de viver mais saudável e sustentável. Essa parcela da população está usando os cogumelos como fonte de proteínas alternativa, que antes eram provenientes de carnes, “Você sabia que o cogumelo pode substituir um bife? Isso porque este fungo tem alto valor proteico [...]” (REVISTA SOCIEDADE DA MESA, 2017).

De acordo com Eira (2012) o alto custo pago pelos cogumelos é um problema para a sociedade e afeta o lucro dos produtores, mas que pode ser combatido com tecnologias de cultivo, fazendo com que o alimento se adapte melhor às condições do país e desta forma torne-se mais acessível a população. A produção de cogumelos também se mostra um negócio sustentável causando um impacto muito pequeno no meio ambiente. Segundo Teixeira (2017) apud Portela (2017) “O melhor de tudo é que esse é um negócio limpo em todos os sentidos: ambiental, ético e socialmente falando”.

A produção de cogumelos também pode ser uma fonte de renda para famílias que vivem em áreas rurais e que possuem pequenas propriedades de terra. “O cultivo de cogumelos no Brasil apresenta-se como uma alternativa viável para pequenos a médios produtores [...]” (EIRA, 2012).

O presente projeto, propõe uma solução de monitoramento e controle para auxiliar os produtores de cogumelos no cultivo deste alimento.

2.1. PROCESSO DE CULTIVO DO AGARICUS BISPORUS

De acordo com Nieuwenhuijnez (2017) o processo de cultivo de cogumelos *Agaricus bisporus*, pode ser dividido nas seguintes etapas:

- produção da semente;
- compostagem;
- pasteurização do composto e inoculação;
- esterilização das estufas de cultivo;

- aplicação da terra de cobertura;
- colheita;

Devido a fatores econômicos, as três primeiras etapas descritas, normalmente são feitas por empresas especializadas, que fornecem o composto de cogumelo inoculado e pronto para cultivo. As etapas seguintes são executadas por produtores, que fazem o cultivo, beneficiamento e também muitas vezes se encarregam das vendas (in natura em conserva).

Alguns dias antes de iniciar o cultivo, deve-se fazer a esterilização das estufas para atenuar riscos de contaminações. Rosa (2017) propõe diversos passos a serem seguidos para realizar a desinfecção das estufas de cultivo, estes passos são descritos a seguir:

1. aquecer a câmara a 28-30°C por 76 horas (4 dias) e colocar a umidade do ar em 60-80%. Este procedimento visa estimular o metabolismo das células latentes, pois assim elas tornam-se mais facilmente destruídas. O agente nocivo pode interferir diretamente no metabolismo celular do microrganismo, durante o crescimento;
2. lavagem da câmara com álcool 70%, por meio de dispersor, no teto, paredes, piso, prateleiras e acessórios internos da câmara. Tempo de atuação 24 horas;
3. lavagem da câmara com hipoclorito de sódio – NaOCl – com 2% de cloro ativo (água sanitária), por meio de dispersor, no teto, paredes, piso, prateleiras e acessórios internos da câmara. Aquecimento da câmara à 75°C por 15 horas e resfriamento gradual até 15°C por cinco horas, com 60-80% umidade (a umidade precisa estar alta para ser efetivo). Tempo de atuação 24 horas. Os esporos do fungo mais resistente (ascósporos de *Byssochlamys fulva*) são destruídos a 90°C;
4. dispersão de formaldeído (formalina) 20%, por meio de dispersor, no teto, paredes, piso, prateleiras e acessórios internos da câmara de cultivo (temperatura 22°C e umidade relativa à 60-80%). Tempo de atuação 24 horas;
5. dispersão de fenol 1,5%, por meio de dispersor, na câmara. Tempo de atuação 24 horas;
6. lavagem com hipoclorito de Sódio – NaOCl - 2% de cloro ativo (água sanitária), com dispersores, no teto, paredes, piso, prateleiras e acessórios internos da câmara de cultivo. Tempo de contato do agente, 24 horas;

7. utilização de filtros microbiológicos em todas as entradas de ar da câmara de cultivo.

Segundo Nieuwenhuijnez (2017) uma vez que o composto de cogumelos, é adquirido e colocado no interior das estufas, sobre prateleiras, é iniciado o processo de desenvolvimento micelial. A temperatura ótima para este desenvolvimento é entre 25 ~ 27°C, e a umidade relativa do ar deve estar em 95% ou mais.

Após 2 semanas é aplicada uma camada de terra de cobertura, que entre outras funções, serve para fornecer bactérias apropriadas e evita que a água seja aplicada diretamente sobre o micélio, o que causaria problemas. Quando a parte mais alta da terra de cobertura é alcançada pelo fungo, deve-se baixar a temperatura da estufa para aproximadamente 20°C.

Ainda segundo Nieuwenhuijnez (2017), os primeiros cogumelos podem ser colhidos em aproximadamente 3 semanas após a aplicação da terra de cobertura e o período de colheita dura em torno de 25 a 30 dias divididos em três fluxos.

2.2. MEDIÇÃO DE TEMPERATURA

Os sensores de temperatura são dispositivos que possuem componentes sensíveis a variações de temperatura, e desta forma são dimensionados para que possam usar esta propriedade para coletar com precisão o valor da temperatura a ser medida.

Nas indústrias são usados diversos tipos de sensores de temperatura, onde cada um apresenta vantagens e limitações. Segundo Albuquerque e Thomazini (2005) os sensores mais utilizados na indústria são os termistores, termopares, termoresistores, par bimetálico, sensores eletrônicos e pirômetros. Cada um dos tipos de sensores apresentam vantagens e desvantagens de acordo com o tipo de aplicação. Os termistores por exemplo, são úteis para a detecção automática, medição e controle de energia física. Já os termopares do tipo T podem ser usados para medir temperaturas abaixo de 0°C, enquanto os termopares do tipo J são recomendados para temperaturas acima de 0°C.

2.2.1. TERMISTORES

Um termistor é um componente que tem sua resistência elétrica alterada com a temperatura. Existem dois tipos de termistores, aqueles com uma resistência que aumenta junto com a temperatura (Coeficiente de Temperatura Positiva – PTC), e

aqueles em que sua resistência decresce com o aumento da temperatura (Coeficiente de Temperatura Negativa – NTC) (HAMPSON, 201-, tradução própria).

Segundo Albuquerque e Thomazini (2005):

São resistores termicamente sensíveis. São semicondutores eletrônicos cuja resistência elétrica varia com a temperatura. Eles são úteis industrialmente para detecção automática, medição e controle de energia física. Os termistores são extremamente sensíveis a mudanças relativamente pequenas da temperatura.

2.2.2. TERMOPARES

De acordo com Goldsmid (2017) o efeito descoberto por Thomas Seebeck em 1821, que leva seu nome, é uma tensão elétrica que aparece quando a junção entre dois condutores diferentes são aquecidos. Este efeito depende da diferença de temperatura, ΔT , entre as duas junções que completam o circuito elétrico. O coeficiente de Seebeck é definido pela equação 1.

$$a_{AB} = \frac{V}{\Delta T} \quad (1)$$

Onde:

a_{AB} = Coeficiente termodinâmico de Seebeck (mV/°C);

V = Diferença de tensão gerada nas extremidades de um condutor (mV);

ΔT = Diferença de temperatura aplicada nas extremidades de um condutor (°C);

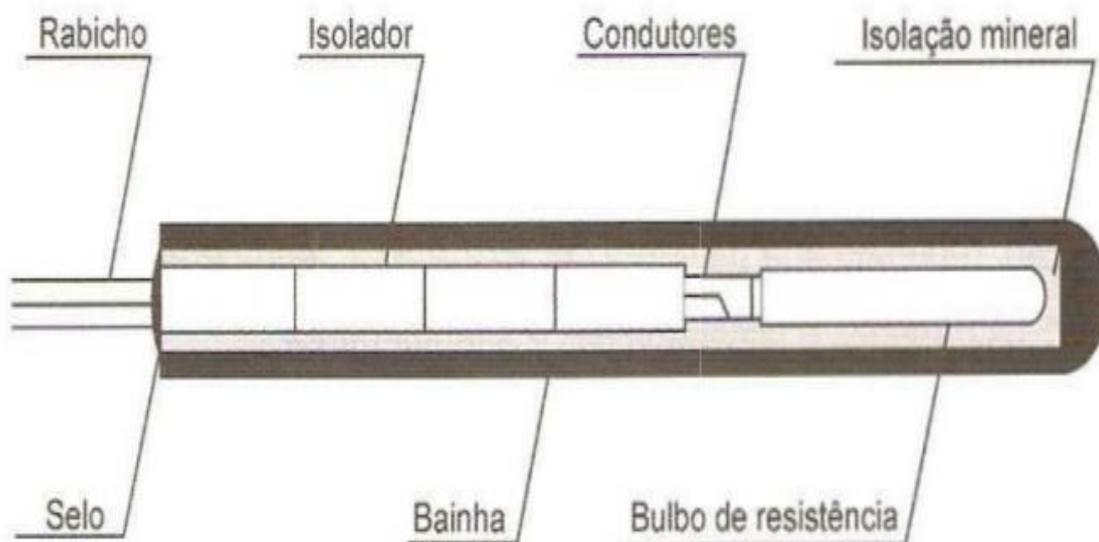
Deste modo, o funcionamento do termopar acontece a partir da medida da diferença de potencial causada por condutores de materiais diferentes em uma dada temperatura. A partir desta diferença de potencial é possível medir diretamente a diferença de temperaturas ou uma temperatura absoluta (ALBUQUERQUE e THOMAZINI, 2005).

2.2.3. TERMORESISTORES

Para Mota (2017) a medição de temperatura feita por termorresistência, tem um coeficiente positivo de temperatura, e seu funcionamento ocorre de acordo com o princípio da agitação térmica nos metais. Sendo o Pt-100 o principal termorresistor, composto de condutores de platina encapsulados em tubos de cerâmica. onde sua resistência a 0°C é de 100Ω (“Pt” descreve o material, que neste caso é a platina).

A figura 1 mostra as principais partes de um termoresistor.

Figura 1 – Termoresistor



Fonte: Albuquerque e Thomazini (2005)

O princípio de funcionamento deste sensor é baseado na alteração da resistência elétrica do material interno em função da variação de temperatura. A medida que a temperatura varia a resistência elétrica do material também varia de forma direta e linear. Essa relação entre a temperatura e resistência é tabelada e deve ser utilizada para que seja possível determinar a temperatura a ser medida (NOVUS, 201-).

2.2.4. PAR BIMETÁLICO

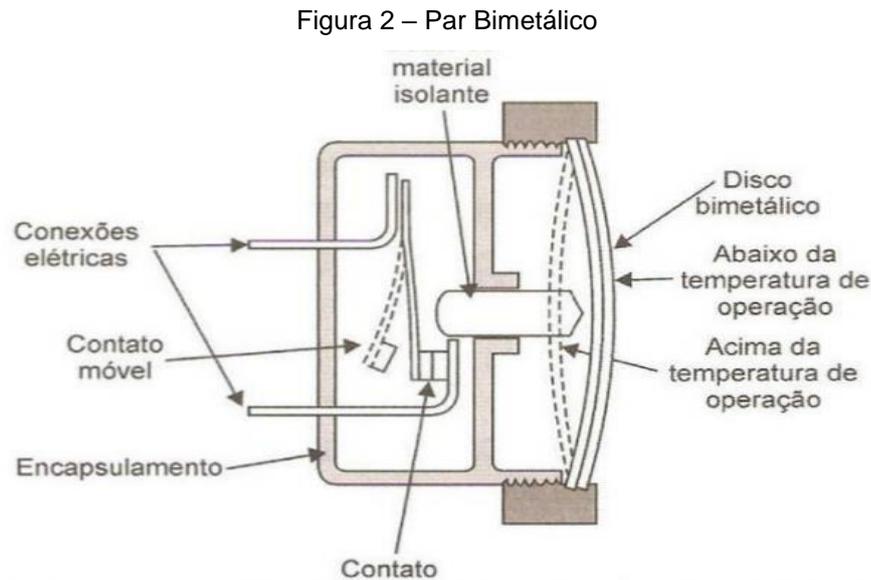
Os termômetros bimetálicos, consistem em lâminas com coeficiente de dilatação linear diferentes, que são combinadas de tal forma a produzir um movimento resultante quando submetidos a uma variação de temperatura. Atualmente este tipo de método de medição, é muito utilizado em termostatos, viabilizando sistemas de controle *on-off* ou sistema de proteção automático contra sobreaquecimentos.

Cada uma das lâminas apresentam uma curvatura que varia em função da temperatura, onde um ponteiro é conectado na lâmina indicando a temperatura correspondente (SEIDEL, 2011).

De acordo com Albuquerque e Thomazini (2005)

Trata-se de duas chapas metálicas de materiais com diferentes índices de dilatação que são montados face a face. Com o aquecimento há uma deformação do conjunto das chapas, havendo a abertura ou fechamento de um contato.

A figura 2 mostra a construção do par bimetálico e o movimento descrito anteriormente é mostrado em linha tracejada.



Fonte: Albuquerque e Thomazini (2005)

2.2.5. SENSORES ELETRÔNICOS

Atualmente existe uma grande variedade de sensores de temperatura eletrônicos que podem ser adquiridos a um custo relativamente baixo, possuem um sinal de saída linearizado e convertido para um protocolo padrão, facilitando a integração com placas de desenvolvimento como o Arduino. Exemplos de alguns sensores eletrônicos são conhecidos pelos códigos LM35, DS18B20, DHT11.

2.2.6. PIRÔMETRO

Pirômetro é um instrumento usado para medir temperaturas sem a necessidade de contato físico, podendo ser classificados em duas classes: pirômetro ótico e pirômetro de radiação. O pirômetro ótico necessita que o corpo emita ondas na região visível do espectro da luz e estar em uma temperatura de no mínimo 750°C até 5500°C. O pirômetro por radiação mede a temperatura por meio da taxa de emissão de energia por unidade de área nas faixas de comprimento de onda relativamente grandes (do visível ao infravermelho curto). Este dispositivo usa um sistema de coleta a energia de um alvo e a focaliza em um detector, que posteriormente será convertida em um sinal elétrico (PIRES et al., 2006).

Os pirômetros de radiação usam a radiação de um corpo para a medida da temperatura sem haver o contato entre o sensor e o corpo. Todos os corpos possuem radiações próprias e contínuas. Pelo aquecimento de um corpo através da

condução de calor ou convecção, uma parte desse calor é transformada em radiação. Uma radiação incidente externa é em parte absorvida, refletida e transmitida pelo corpo. A soma dos coeficientes de absorção, reflexão e transmissão é igual a 1 (ALBUQUERQUE e THOMAZINI, 2005).

2.3. MEDIÇÃO DE UMIDADE RELATIVA DO AR

A atmosfera é composta por diversos gases diferentes, sendo o vapor de água um deles e talvez o mais importante. A umidade absoluta do ar, expressa em gramas por m³, é a quantidade real de água contida em um conhecido volume de ar. Geralmente essa quantidade de água é expressa como umidade relativa (UR), é a razão entre a quantidade de umidade presente no ar, e a quantidade máxima de vapor de água que o ar pode conter em uma determinada temperatura (ALBUQUERQUE e THOMAZINI, 2005).

Para o processo de cultivo dos cogumelos, é essencial o monitoramento da umidade relativa do ar no interior das estufas. “Outro fator importante no que diz respeito ao desenvolvimento micelial é a presença de suficiente umidade. Por conseguinte, a umidade relativa (UR) deve ser muito elevada (UR de 95% ou ainda mais elevada)” (NIEUWENHUIJNEZ, 2007) e também de acordo com Albuquerque e Thomazini (2005)

A umidade é um importante parâmetro de muitos processos industriais, uma vez que o vapor d'água contido no ar, ou em um gás, pode alterar as especificações de um produto ou serviço afetando a sua qualidade e, conseqüentemente, a sua aceitação pelo mercado.

Para selecionar o correto instrumento de medição de temperatura, existem muitas variáveis que precisam ser avaliadas no local de instalação deste instrumento. De acordo com Rogers (2015) algumas questões a serem verificadas para a durante a fase de especificação dos sensores de umidade são:

- Qual o range de umidade a ser medido?
- Qual é o range de temperatura do ambiente?
- Existe muita umidade no local ou o sensor poderá condensar?
- Haverá pouca umidade no local?
- Existem elementos químicos ou contaminantes neste local?
- Qual é a precisão necessária para o processo?

O sensor especificado para este projeto está susceptível a problemas com condensação do ar no interior da sonda. De acordo com o fabricante, a condensação do ar no interior da sonda pode causar avarias, e comprometer o funcionamento do mesmo (AOSONG, 201-).

Para evitar este problema é necessário manter a temperatura do ponto de orvalho abaixo da temperatura do ambiente. Desta forma será necessário tomar cuidado com oscilações de temperatura no interior da estufa já que em quase todo o processo a umidade relativa do ar está bem alta, e também no momento da redução de temperatura (momento em que o cogumelo passa da fase de desenvolvimento micelial para a fase de frutificação). “Se for possível, dever-se-á reduzir a temperatura do composto, em 6°C, até 20°C, dentro de 6 dias (valores aproximados)” (NIEUWENHUIJNEZ, 2007).

Outro cuidado a ser tomado, de acordo com Albuquerque e Thomazini (2005) é a respeito de agentes contaminantes (químicos e partículas)

Todos os sensores de umidade são susceptíveis à contaminação, cujos níveis variam muito com a aplicação. Isso diminui a precisão, aumenta o tempo de resposta e encurta a presumível durabilidade. Certos sensores são definitivamente danificados pela contaminação. Outros podem ser limpos ou reativados. Mas todos são afetados, em maior ou menor grau, pela constante presença de contaminantes nos processos industriais.

O sensor especificado para este projeto, possui um filtro contra contaminação por partículas, mas não se sabe o quanto efetivo isto é, e desta forma, para manter o bom funcionamento do instrumento será necessário tomar cuidados durante os processos de limpeza do ambiente devido ao uso de produtos químicos (produtos de limpeza) e possíveis contaminações por poeira e micro partículas.

Uma boa prática é proteger o sensor com plástico ou outro tipo de proteção durante a limpeza, e durante o processo de cultivo, manter o ambiente o mais limpo possível.

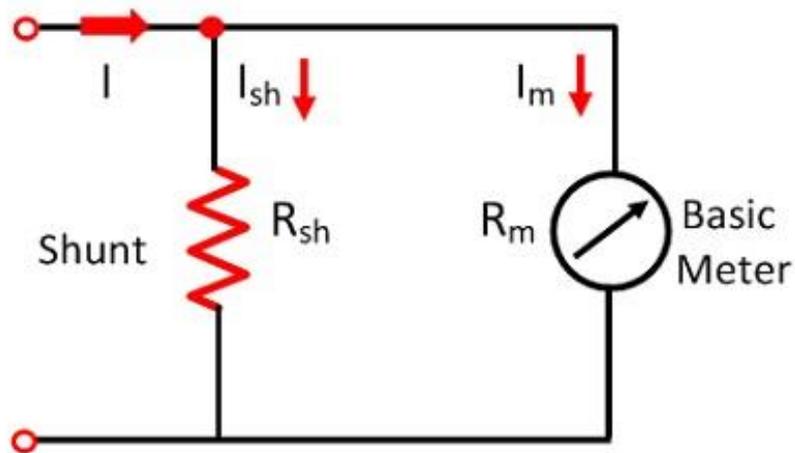
2.4. MEDIÇÃO DE CORRENTE ELÉTRICA

2.4.1. MEDIÇÃO DE CORRENTE CONTÍNUA POR RESISTOR SHUNT

Uma das formas de medir a corrente elétrica, contínua ou alternada, é por meio de um resistor *shunt*. Neste método é usado uma resistência de valor muito baixo, colocada em paralelo com o circuito que drena a corrente a ser medida. Deste modo, medindo a queda de tensão entre os terminais do dispositivo, pode-se obter a

corrente do circuito (ALBUQUERQUE e THOMAZINI, 2005). Na figura 3 é mostrado um esquemático indicando como é feita a medição de corrente com resistor *shunt*.

Figura 3 – Medição de Corrente por Resistor Shunt



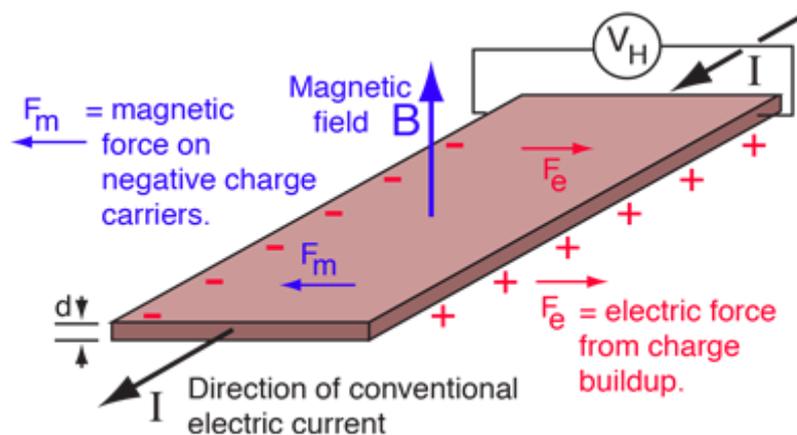
Fonte: CIRCUIT GLOBE, 201-

2.4.2. MEDIÇÃO DE CORRENTE CONTÍNUA POR EFEITO HALL

Um outro método de medição da corrente elétrica contínua e alternada, é pelo efeito hall. Esse efeito foi descoberto pelo físico americano Edwin Herbert Hall (1855-1938) em outubro de 1879 na Universidade Johns Hopkins nos Estados Unidos. Ele realizou um experimento no qual foi observado o efeito que hoje é conhecido como Efeito Hall Clássico (EHC). Estes estudos foram publicados em 1879 (American Journal of Mathematics², p. 287) e em 1880 (Philosophical Magazine 9, p. 225) (BASSALO, 2015).

A figura 4 pode ser usada como base para explicar o funcionamento do efeito hall.

Figura 4 – Efeito Hall



Fonte: HYPER PHYSICS, 201-

Segundo Albuquerque e Thomazini (2005)

Os sensores de Hall são constituídos de dispositivos semicondutores que sofrem influência de campo magnético, que faz gerar nas laterais da placa condutora uma diferença de potencial (Lei de Lorentz) o qual se conecta um circuito de medição [...].

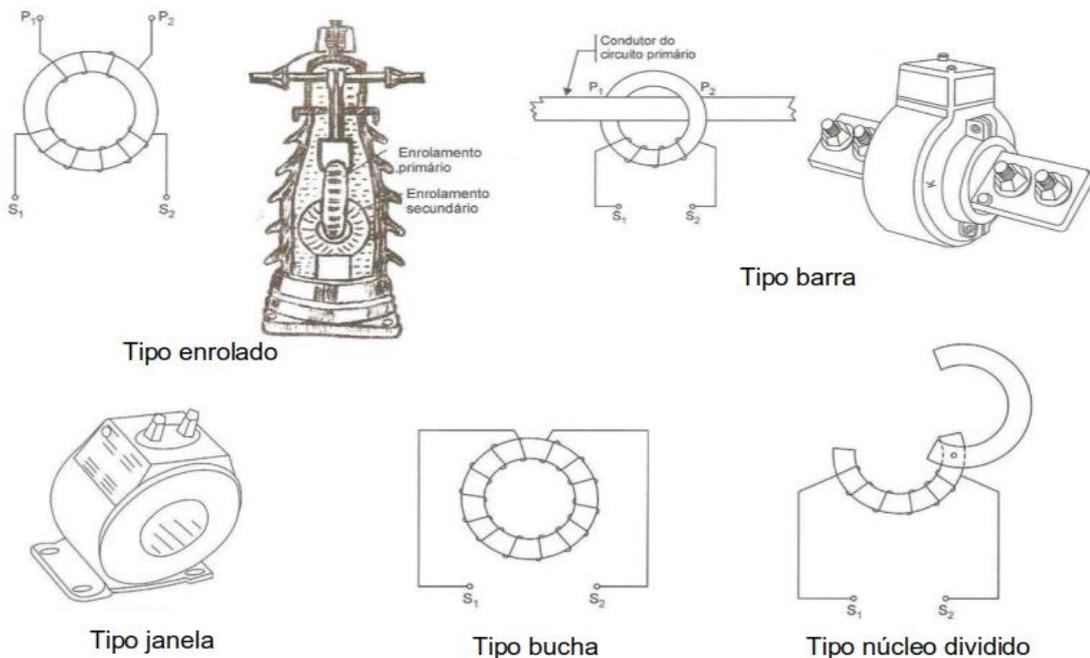
Ainda de acordo com Albuquerque e Thomazini (2005) uma vantagem deste método de medição, é a possibilidade de fazer a medida de campos magnéticos gerados por correntes elétricas contínuas ou alternadas, com o mesmo dispositivo.

2.4.3. MEDIÇÃO DE CORRENTE ALTERNADA POR MEIO DE TC

Atualmente nas indústrias, é muito comum realizar a medição de corrente elétrica alternada, usando os dispositivos de medição chamados transdutores de corrente. “Os transdutores de tensão ou corrente tem como finalidade transformar sinais de tensão ou corrente em sinais de tensão ou corrente DC, permitindo sua indicação em CLPs e indicador digitais.” (KRON, 2010).

Os autores Albuquerque e Thomazini (2005) categorizaram os transdutores de corrente em 5 tipos: os transdutores do tipo enrolado, tipo barra, tipo janela, tipo bucha e transdutor de corrente de núcleo dividido. Estes transdutores são mostrados na figura 5.

Figura 5 – Transdutores de Corrente



Fonte: Albuquerque e Thomazini (2005)

2.5. GRAU DE PROTEÇÃO IP – NORMA IEC60529

A norma IEC60529:1989 (primeira versão desta norma), criada pela IEC – Comissão Internacional de Eletrotécnica, surgiu para classificar os padrões de proteção contra a entrada de partículas sólidas e água no interior de invólucros de equipamentos, sensores, dispositivos, entre outros exemplos. Ela também define os métodos de ensaios para a verificação desse nível de proteção.

De acordo com Zafalon (201-)

Essas normas definem métodos de ensaios para verificação do nível de proteção às partes perigosas (contato com partes vivas de baixa ou alta tensão e contato com partes mecânicas perigosas), contra a penetração de objetos sólidos estranhos e/ou contra a penetração de água e o sistema de codificação para indicar os graus de proteção providos por um invólucro.

Por definição, a numeração do grau de proteção, é indicado nos catálogos do produto, sempre precedido das letras “IP” (do inglês, *Ingress Protection*), e precedido de dois números, sendo o primeiro a indicação da proteção contra as partículas sólidas, e o segundo indica a proteção contra a entrada de água.

A figura 06 mostra um exemplo de placa de motor onde está indicado o grau de proteção IP55 ao lado esquerdo da placa.

Figura 6 – Exemplo de informações técnicas – grau de proteção

INMETRO **ENERGIA** Reservatório térmico
Fornecedor **Pro - Sol**
Marca **Pro - Sol**
Modelo **RT 200L AP - S/ RESISTÊNCIA**
Aplicação **Banho**

Volume (l)	200,0	Data fabricação	
Potência (W)	N/A	Pressão máxima de trabalho	197 (kPa) 20,1 (mca)
Corrente (A)	N/A	Tensão (V)	N/A
Disjuntor (A)	N/A	Frequência (Hz)	N/A
Seção transversal mínima dos condutores de alimentação (mm ²)	N/A	Grau de proteção	IP24

PROCEL **PROGRAMA BRASILEIRO DE ETIQUETAGEM** **Segurança**
MHC Registro de Tecnologia 00P 0093 Registro Nº 002 342/2017

Instruções de instalação e recomendações de uso, leia o manual do aparelho.

A proteção indicada pelos números é explicado a seguir:

Primeiro Dígito:

- 0 – Não protegido;
- 1 – Proteção contra objetos sólidos com 50mm de diâmetro ou mais;
- 2 – Proteção contra objetos sólidos com 12,5mm de diâmetro ou mais;
- 3 – Proteção contra objetos sólidos com 2,5mm de diâmetro ou mais;
- 4 – Proteção contra objetos sólidos com 1mm de diâmetro ou mais;
- 5 – Proteção contra poeira;
- 6 – À prova de poeira.

Segundo Dígito:

- 0 – Não protegido;
- 1 – Protegido contra gotas que caem na vertical;
- 2 – Protegido contra gotas que caem na vertical, com corpo inclinado em 15°;
- 3 – Protegido contra borrifo de água;
- 4 – Protegido contra jorro de água;
- 5 – Protegido contra jatos de água;
- 6 – Protegido contra jatos potentes de água;
- 7 – Protegido contra a imersão temporária em água de até 1 metro por 30 minutos;
- 8 – Protegido contra a imersão contínua em água.

No Brasil, em 2001, foi criada a norma ABNT NBR IEC 60529:2001 (corrigida em 2005 e substituída em 2017 pela ABNT NBR IEC 60529:2017), criada com base na norma internacional IEC 60529.

A aplicação desta norma pode ajudar a definir uma qualidade mínima desejada em produtos e equipamentos, aumentar a confiabilidade e também a segurança. O autor Zafalon (201-) esclarece que um produto protegido é capaz de evitar que as pessoas sofram danos físicos, como choque elétrico, ferimentos causados por partes móveis, e também evitar danos ao próprio equipamento [...].

2.6. BANCO DE DADOS

Um banco de dados pode ser definido como um conjunto de dados armazenados com um propósito específico (FERNANDES, 2017).

São exemplos de bancos de dados: cadastro de produtos em um supermercado, cadastro de clientes em um banco, cadastro de livros em uma biblioteca.

Os bancos de dados podem ser classificados em dois tipos, os relacionais e os não relacionais. Em um banco de dados relacional, os dados são organizados em tabelas, com relações entre elas. Este arranjo de tabelas, permite que dados sejam extraídos de forma mais rápida quando comparados aos bancos de dados não-relacionais (FERNANDES, 2017). São exemplos de banco de dados relacionais: *SQL Server, MySQL, PostgreSQL* (OPSERVICES, 2016).

Para a implementação dos bancos de dados relacionais, é usada a linguagem padrão SQL (*Structured Query Language*), que tem a capacidade de executar ações, além de outras coisas, como inserir, procurar, atualizar ou deletar informações no banco de dados (GURU99, 201-).

Um banco de dados não relacional, é um banco que armazena dados de forma sequencial. Eles são usados desde os anos 60, e a principal vantagem deste tipo de banco é que são melhores para armazenar dados não estruturados, como por exemplo: e-mails, dados multimídia e dados provenientes de mídias sociais (FERNANDES, 2017).

O conceito deste tipo de banco de dados cresceu com empresas como Google, Facebook, Amazon, etc. Pois elas têm a necessidade de manipular grandes volumes de dados. Quando é usado um banco de dados relacional para armazenar estes grandes volumes, o desempenho dos bancos cai, tornando muito lentos (GURU99, 201-).

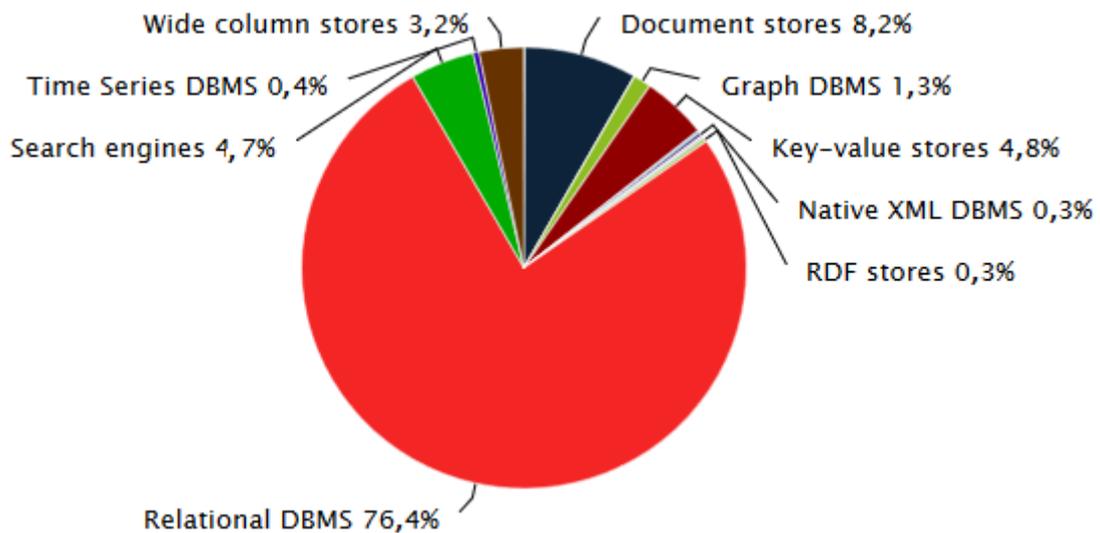
Atualmente são exemplos de banco de dados não relacionais: MongoDB, Cassandra, Redis, HBase, Amazon DynamoDB, Neo4j (MATOS, 2018).

Nos dias de hoje, com o advento da internet das coisas e a chamada indústria 4.0, onde a quantidade de dispositivos conectados na rede cresce rapidamente, a forma de gerenciar e armazenar as informações produzidas por estes dispositivos, também precisa se readequar para acompanhar essa demanda (FERNANDES, 2017). Os bancos de dados não relacionais estão crescendo, pois em algumas aplicações específicas, modelos de banco de dados baseado em grafos são melhores como por exemplo o Twitter: se uma pessoa começa a seguir outra, a segunda não necessariamente precisa seguir a primeira, pois isso não é uma relação bidirecional (TARDIVEL, 2017).

No site <https://db-engines.com/en/ranking>, é possível conferir quais bancos de dados são os mais usados atualmente. De acordo com esse site, 76,4% dos bancos

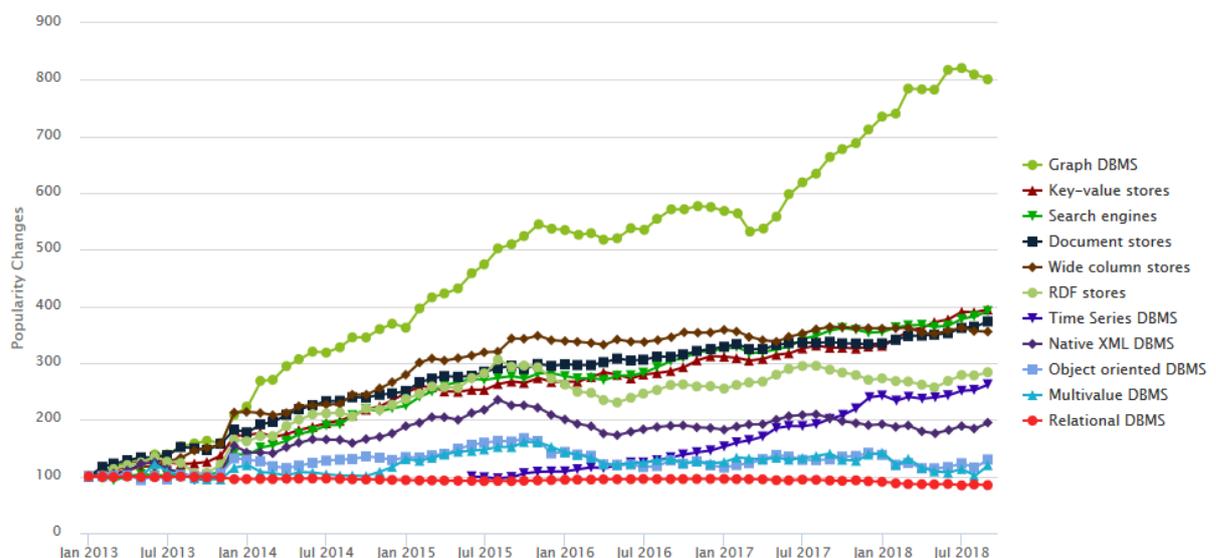
de dados usados hoje são bancos do tipo relacional conforme mostrado na figura 7. Entretanto, o uso dos bancos de dados não relacionais vem crescendo rapidamente, conforme mostrado na figura 8.

Figura 7 – Porcentagem do uso de banco de dados por tipo



Fonte: DB-ENGINES, 2019

Figura 8– Tendência do uso de banco de dados por tipo



Fonte: DB-ENGINES, 2019

2.7. LINGUAGEM DE PROGRAMAÇÃO PYTHON

Python é uma linguagem de programação livre, de alto nível, orientada a objetos e interpretada, que atualmente está na sua versão 3.7.0. Em meados de 1989, foi dado início ao desenvolvimento desta linguagem de programação, que teria sua primeira versão (0.9.0), sendo lançada no dia 20 de fevereiro de 1991. O criador dela é um programador holandês chamado Guido Van Rossum (MAGNUN, 2014),

nascido em 31 de janeiro de 1956, formado em Matemática na *University of Amsterdam* (REVISTA PROGRAMAR, 2018).

Em 1982 Guido trabalhou na CWI (*Centrum Wiskunde & Informatica*), em Amsterdã, Holanda, e em 1987 foi transferido para o grupo de trabalho “*Amoeba*” liderado por Andrew Tanenbaum. Neste grupo de trabalho Guido percebeu a necessidade de desenvolver uma linguagem de programação intermediária entre o C e o *ShellScript*, que foi o principal objetivo da implementação do Python (MAGNUN, 2014).

Atualmente, o nome *Python*, está associado à cobra Píton, mas este nome surgiu baseado na série de TV “*Monty Python’s Flying Circus*”, que segundo o seu criador, tinha como objetivo evitar um nome pouco impactante como siglas, e um nome que não tivesse um significado profundo (MAGNUN, 2014).

A linguagem de programação *Python* é recomendada para iniciantes quando se trabalha com a *Single Board Computer (SBC) Raspberry Pi*, um dos motivos é pela simplicidade e facilidade de compreensão quando comparadas a outras linguagens de programação como C ou C++ (MAGNUN, 2014).

2.8. LINGUAGEM DE PROGRAMAÇÃO JAVA SCRIPT

A linguagem de programação hoje conhecida como *JavaScript*, inicialmente chamada de *Live Script*, foi desenvolvida pela empresa *Netscape Communications Corporation*, que tem como seu criador Brendan Eich (LIMA, 2006).

Em 1994, o *Live Script* foi criado com o propósito de trabalhar com aplicações nas páginas HTML (*Hypertext Markup Language*), dando função de interatividade, na época desenvolvida para o já extinto navegador *Netscape 2.0* e também para o *Internet Explorer 3.0* (LIMA, 2006).

No ano de 1994, a empresa *Netscape* estava em conflito com a *Microsoft* em um episódio conhecido como a “guerra dos navegadores”, onde as duas empresas disputavam diretamente a venda do software utilizado para busca de informações na internet (DISCOVERY CHANNEL, 2013). Em uma ação estratégica a *Netscape* firmou uma parceria informal com a *Sun Microsystems*, objetivando parcerias de troca de tecnologias e distribuição de produtos, tentando ao máximo competir com a gigante *Microsoft* (O’ SHAUGHNESSY, 2017).

Devido a esta parceria, e com influência da *Sun Microsystems*, a linguagem de programação *Live Script* mudou para *Java Script*, pois a *Sun* já havia desenvolvido a linguagem *Java*. Mas apesar da similaridade nos nomes, *Java* e *JavaScript* são coisas diferentes e com propósitos diferentes, pois *Java* é usado para o

desenvolvimento de aplicações, enquanto o *JavaScript* está embutida nas páginas HTML e são interpretadas pelo navegador (O' SHAUGHNESSY, 2017).

2.9. REDES DE COMPUTADORES

De acordo com Tanenbaum (2011) o grande número de implementações das redes de computadores indica que elas têm atraído o interesse das pessoas. Uma das vantagens mencionadas, é o compartilhamento de informações (registros de clientes, estoques, contas, extratos, etc.) e recursos independente de sua localização física (impressoras, *scanner*, etc.).

2.10. PROTOCOLO ETHERNET

Durante os anos 70, as empresas Xerox, Intel e *Digital Equipment Corporation* (DEC), desenvolveram um protocolo de comunicação para redes locais, desenvolvida inicialmente como uma solução de interconexão de equipamentos para escritórios.

Esta rede foi desenvolvida com base em uma outra rede chamada de ALOHA, criada na década de 60 pelo engenheiro americano Norman Abramson, da Universidade do Havaí (GALLO, 2003).

Na figura 9, é mostrada uma foto que mostra a primeira transferência de dados feita usando a rede ALOHA.

Figura 9 – Primeira transferência de um pacote de dados via rádio sobre a ALOHAnet.



Fonte: UNIVERSITY OF HAWAII, 2019

Os trabalhos realizados pelas empresas Xerox, Intel e *Digital Equipment Corporation* (DEC), resultaram em um padrão chamado de *Ethernet* Versão 2.0, publicado em 1982. Mas mesmo assim, esse padrão ainda não era admitido como um padrão internacional para redes locais. Subsequentemente a IEEE produziu uma especificação padrão desenvolvida com base na *Ethernet* V2.0, conhecido como padrão IEEE 802.3.

Apesar destes dois padrões serem similares, eles não são compatíveis, havendo diferenças como: tamanho do cabo, função do transceptor, formato dos quadros e topologia (GALLO, 2003).

De acordo com Carissimi (2009) a concepção da rede *Ethernet* obteve êxito e serviu como base para o desenvolvimento de um padrão conhecido como IEEE 802.3, destinado para redes locais. Devido à similaridade dos padrões, atualmente é comum denominarmos as redes IEEE 802.3 como redes *Ethernet*.

No princípio, os cabos utilizados para compor a rede eram cabos coaxiais de meia polegada, combinados com dispositivos de rede chamados de *tranceptors*. Posteriormente, com o intuito de reduzir custos, foram desenvolvidos cabos coaxiais alternativos, conhecidos como *Ethernet* de Fio Fino (*THIN-wire ethernet* ou *thinnet*).

Thinnet é um cabo mais fino, mais barato e mais fácil de instalar (com o uso de conectores BNC, eles podiam ser facilmente conectados e desconectados, sem o uso de ferramentas), mas que também possui desvantagens, como ser mais susceptível a interferência elétrica (que não era necessariamente um problema em escritórios), opera em distâncias menores e suporta menor número de conexões de rede.

Com o avanço tecnológico, os cabos coaxiais deixaram de ser usados, e hoje praticamente só se encontram redes com cabos UTP (*Unshielded twisted Pair*), que permitem que um computador (ou outro dispositivo), acesse a rede utilizando um cabo de par metálico sem blindagem, reduzindo os custos de implantação das redes locais (COMER, 2006).

3. DESENVOLVIMENTO

A primeira etapa do projeto será o levantamento do processo, as características do local de instalação do sistema e funcionalidades desejadas do sistema. Em seguida será feita a especificação dos equipamentos/componentes (elétricos, eletrônicos, instrumentos) usados no projeto, verificando as características mínimas exigidas pelo processo em questão, levando em conta fatores como: o local de instalação, as condições de operação, custo, estabilidade.

Uma vez especificados os equipamentos/componentes usados no projeto, o próximo passo é desenvolver os esquemáticos para o projeto eletrônico e diagrama unifilar do projeto elétrico, desenvolvendo todos os cálculos necessários e dimensionamentos, para em seguida fazer a aquisição dos componentes.

Para o projeto eletrônico, uma vez tendo o esquemático e os cálculos, é possível realizar a montagem em matriz de contatos e então realizar testes para validar o funcionamento. Uma vez comprovado o funcionamento do projeto então será dado início a montagem da placa eletrônica.

Para o projeto do painel elétrico, com o diagrama unifilar pronto, é possível passar para o próximo passo indicando a disposição dos componentes no interior da caixa metálica, calculando a taxa de ocupação dos cabos no interior das canaletas de plástico, espaçamentos necessários e os demais dimensionamentos necessários para realizar a montagem. Após isso será necessário fazer a aquisição dos componentes e ferramentas usadas para efetuar a montagem do painel.

A etapa do desenvolvimento de software pode ocorrer de forma paralela, sendo iniciada pela escolha das linguagens de programação, e então o desenvolvimento dos códigos da unidade de controle eletrônica, da interface de comunicação e a linguagem do banco de dados, que serão testados na fase do estudo de caso.

A última parte consiste em fazer a montagem em campo e realizar um estudo de caso para testar as funcionalidades propostas no projeto.

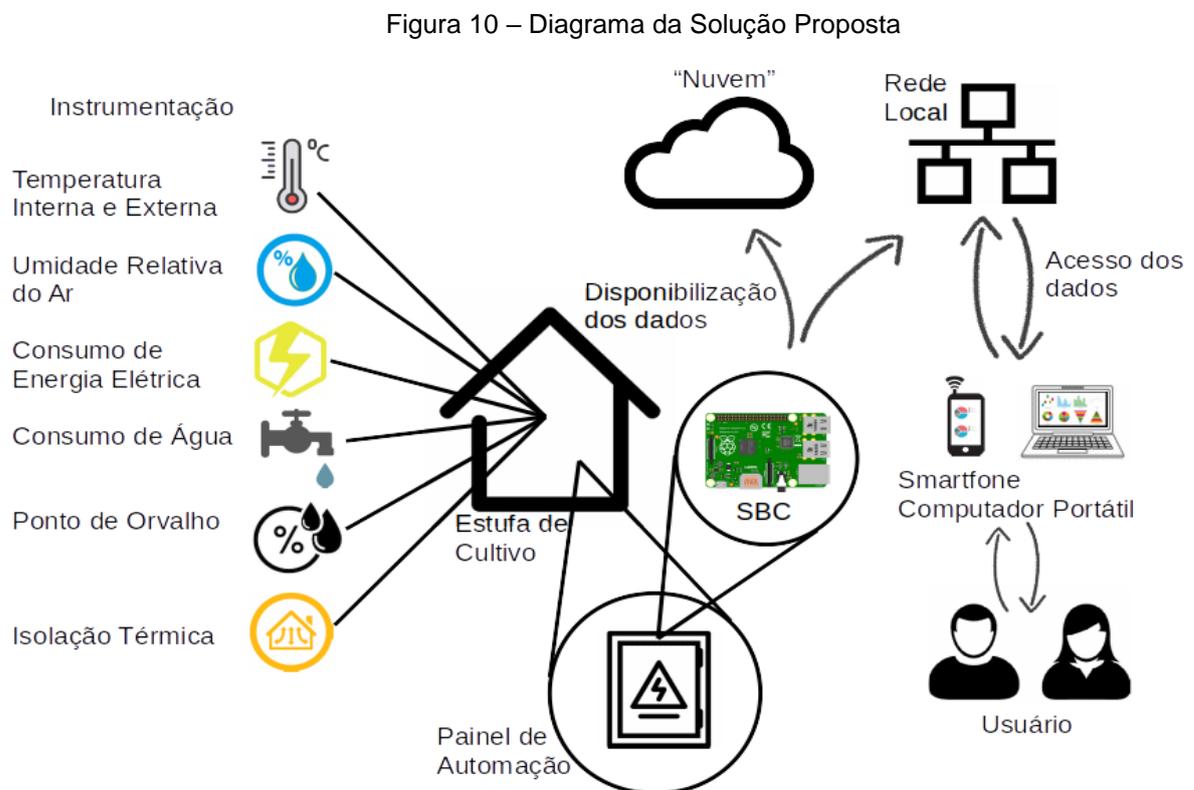
3.1. PROPOSTA GERAL DO PROJETO

O projeto é constituído de instrumentos instalados no interior e exterior da sala de cultivo, monitorando variáveis que influenciam na produção. Estes dados são lidos por uma *Single Board Computer*, que processa essas informações, organiza e disponibiliza elas, em forma de gráficos e relatórios, em uma rede local (com possibilidades futuras dessas informações serem disponibilizadas em nuvem).

O fungicultor pode acessar essas informações por meio de um *smartphone* ou computador com acesso à essa rede local, usando um navegador de internet. É necessário saber o endereço IP da SBC conectada na rede local, para pode enviar uma requisição TCP/IP, que é respondida por uma página HTML contendo as informações relevantes ao seu cultivo.

Também será possível dar alguns comandos como: habilitar o funcionamento do condicionador de ar, ligar ou desligar o sistema admissão de água no interior da estufa que faz o controle da umidade relativa do ar, ou ligar ou desligar o sistema de exaustão de ar para o interior da estufa.

A figura 10, mostra um diagrama da solução proposta.



Fonte: 2019, Autor

3.2. AVALIAÇÃO DO PROCESSO DE CULTIVO

A etapa do processo de cultivo foi abordada no capítulo 2.1 “Processo de Cultivo do *Agaricus bisporus*”. Este projeto se limitou a monitorar as seguintes variáveis: temperatura e umidade interna da sala de cultivo, a temperatura e umidade do ambiente externo, nível de CO₂ no interior da sala de cultivo, consumo de energia e ocupação de pessoas. Existem outras variáveis que são importantes para o processo de cultivo (como por exemplo pH e temperatura do composto) que

não foram incluídas nesta etapa do trabalho, mas que podem ser trabalhadas em uma etapa futura.

É previsto que a temperatura interna da sala de cultivo irá variar em um intervalo de 17°C até no máximo 30°C durante o processo de cultivo, mas a temperatura máxima pode chegar a 90°C durante a sanitização do ambiente, enquanto a umidade relativa do ar interna a sala de cultivo deve ser controlada entre valores que variam de 85% até 95% durante o processo de cultivo e valores mais baixos durante o processo de sanitização. Já a previsão do intervalo da temperatura e umidade externa depende da região em que o projeto se encontra instalado. O intervalo do nível de CO₂ no interior da sala de cultivo depende das dimensões e da quantidade de composto mas pode chegar a valores superiores a 5000ppm, e é produzido pelo fungo devido à sua atividade de colonização do composto.

Para o processo de cultivo é uma vantagem que essas variáveis sejam lidas por sensores, armazenadas em um sistema e disponibilizadas de forma remota. Isso evita que o produtor tenha que entrar na sala de cultivo para fazer a leitura dos sensores o que reduz a probabilidade de levar contaminação do exterior para o interior da sala de cultivo, além de reduzir a necessidade de mão de obra. Ainda com relação ao uso dos sensores, eles viabilizam a execução de várias leituras ao longo do dia (esse projeto prevê 12 medições por hora), o que é impraticável quando feito de forma manual.

Assim como qualquer outro produto, para que a os cogumelos produzidos sejam uma atividade rentável, é importante que várias informações sejam contabilizadas como: insumos, fornecedores, perdas, rendimento, etc. Deste modo esse projeto prevê armazenar o maior número de informações referentes à cada safra cultivada, em um banco de dados, de tal forma que essas informações possam ser acessadas e utilizadas como base, no cálculo de indicadores (custo de produção, rendimento).

3.3. SELEÇÃO DOS INSTRUMENTOS E EQUIPAMENTOS

3.3.1. SINGLE BOARD COMPUTER - SBC

O sistema de monitoramento proposto depende de uma unidade eletrônica de controle, que será instalada em campo e terá a função de receber e armazenar as variáveis de processo coletadas pelos sensores instalados na sala de cultivo, e deverá repassar essas informações para o usuário por meio de uma interface de comunicação.

A SBC utilizada no projeto deve no mínimo atender as seguintes características:

- Possuir uma interface Ethernet e ter a capacidade de operar como servidor de uma rede local TCP/IP respondendo às requisições HTTP dos dispositivos conectados em uma rede local;
- Portas USB, para transferência dos dados coletados e conectividade de dispositivos periféricos;
- Acesso remoto via protocolo de rede SSH;
- Interface de comunicação *Serial RS485 - Modbus RTU*;
- Temperatura de operação entre -10°C ~ 50°C ;

A SBC selecionada para este projeto foi o *Raspberry Pi 3 – Model B*, pois atende os requisitos mencionados anteriormente (com exceção da temperatura de operação, que será melhor discutido mais adiante) e possui um custo relativamente barato. Apesar de existirem muitas alternativas superiores a esta SBC, alguns fatores extras ajudaram na tomada de decisão, um deles é a quantidade de informações disponíveis sobre o *Raspberry Pi* na *internet*, com muitos exemplos de códigos e bibliotecas dedicadas, trabalhar com esta SBC não é algo tão desafiador. Outro fator importante, são os modelos de caixas para montagem específicos para esta eletrônica, a empresa *Hitaltech* por exemplo, disponibiliza vários modelos de caixas compatíveis com o *Raspberry Pi 3 Model B*, e que podem ser montados em trilho DIN de 35mm. Como a intenção é colocar a eletrônica dentro de um painel de automação, isso facilita bastante a montagem no interior do painel.

Um problema com esta placa está na temperatura de operação, pois o fabricante recomenda uma faixa de trabalho de 0°C até 45°C , especificação inferior à dos concorrentes que normalmente são entre $-10^{\circ}\text{C}/-20^{\circ}\text{C}$ até 50°C . Na região sul do Brasil, na época de inverno é bem comum registrar temperaturas negativas em algumas localidades, e no verão a temperatura máxima pode chegar muito próximo de 35°C , dentro do painel metálico se o mesmo for exposto ao sol, é bem provável que a temperatura chegue próximo do seu limite de operação e isso talvez ocasione problemas ao bom funcionamento da eletrônica.

Com base nessa informação, o painel de automação que irá abrigar esse componente, será instalado no lado de dentro das estufas de cultivo, para evitar que a haja algum problema relacionado com alta temperatura. Entretanto com o painel no interior da estufa não será possível aplicar o método de sanitização mencionados no capítulo “2.1. Processo de Cultivo do *Agaricus bisporus*” item 3.

O acesso remoto à placa via protocolo de rede SSH facilita a atualização dos códigos, alterações e futuras manutenções no sistema. Com um computador portátil é possível acessar o sistema operacional da SBC sem a necessidade de retirar ela do painel, parar o processo ou abrir a porta do painel, reduzindo assim também o tempo de manutenção, o que traz uma grande vantagem para o projeto.

A figura 11 mostra uma imagem do Raspberry Pi 3 Model B.

Figura 11 – Raspberry Pi 3 Model B



Fonte: RASPBERRY PI, 2019

3.3.2. PAINEL DE AUTOMAÇÃO

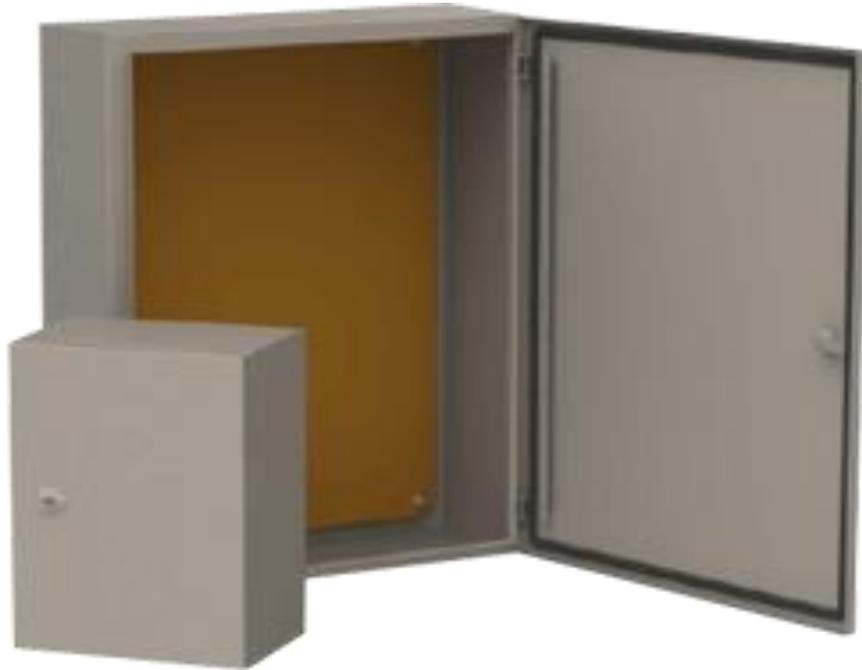
O painel metálico que irá comportar os componentes do sistema de monitoramento será instalado em um ambiente muito úmido e que pode ser eventualmente submetido a lavagens não intencionais. Desse modo o painel deve possuir um grau de proteção mínimo IP65.

Existem diversos modelos comerciais deste tipo de painel metálico, e o painel escolhido possui uma vedação de borracha, chapa com tratamento de fosfato de zinco e pintura a pó, nas dimensões de 600mmx400mmx250mm, placa de montagem na cor laranja, e pintura externa na cor bege.

Ainda com relação ao painel de automação, é desejável que o mesmo possua uma flange inferior removível, pois isso facilita bastante a montagem dos eletrodutos conectados ao painel.

A figura 12 mostra uma imagem da caixa metálica para montagem do painel de automação.

Figura 12 – Caixa Metálica Painel de Automação



Fonte: BSE PAINÉIS, 2019

3.3.3. SENSOR DE TEMPERATURA E UMIDADE

Os sensores de temperatura e umidade instalados nas salas para cultivo de cogumelos, têm a função de realizar medições que podem ser utilizadas para o controle do sistema de aquecimento e resfriamento e também para o monitoramento da temperatura e umidade.

Como existem diversos tipos de sensores de temperatura e umidade, essa análise comparativa tem o objetivo de selecionar qual é o tipo de sensor mais adequado para esta aplicação.

A tabela 1, mostra os tipos de sensores envolvidos nesta análise, bem como as características consideradas. É desejável selecionar um sensor que no mínimo funcione em um intervalo de umidade de 85% até 100%, intervalo de temperatura de 17°C até 60°C, tenha uma precisão compatível com o processo (temperatura +- 1°C e umidade relativa +-5%, ou melhor).

Tabela 1 – Tipos de sensores de temperatura e umidade

Instrumento	Intervalo Umidade [%]	Intervalo Temperatura [°C]	Precisão Umidade	Precisão Temperatura
Aosong DHT11	20% a 90%	0°C a 50°C	±5% ¹	±2°C
Aosong DHT22	0% a 100%	-40°C a 80°C	±2% ¹	±0,5°C
Aosong AM2315	0% a 100%	-40°C a 125°C	±2% ¹	±0,1°C
Liohm FHT20	0% a 100%	-20°C a 80°C	±3% ¹	±0,3°C
Sigma EE60	0% a 100%	0°C - 50°C	±2,5% ¹	±0,3°C

Instrumento	Estabilidade Umidade [%/ano]	Estabilidade Temp. [°C/ano]	Interface de Saída	Custo [R\$]
Aosong DHT11	±1% p/ Ano	?	<i>Serial – Single Wire Two Way</i>	R\$10,00
Aosong DHT22	±0,5% p/ Ano	?	<i>Serial – Single Wire Two Way</i>	R\$17,90
Aosong AM2315	±0,5% p/ Ano	±0,1°C p/ Ano	I2C	R\$68,16 ²
Liohm FHT20	±0,5% p/ Ano	±0,1°C p/ Ano	4-20mA / 0 a1V 0 a 5V / 0 a 10V	R\$589,00
Sigma EE60	?	?	0 a1V / 0 a 5V 0 a 10V	R\$1956,00

Fonte: 2019, Autor

Notas:

1. Esse valor refere-se ao intervalo de 20% a 80%. Essa precisão não é constante e depende do intervalo de operação;
2. Esse sensor não é vendido no Brasil, e deve ser importado. Esse valor não inclui as taxas de transporte e impostos;

De acordo com a análise, pode-se eliminar o sensor DHT11 do fabricante AOSONG, pois o mesmo não opera em ambientes com umidade superior a 90%.

O sensor EE60 do fabricante Sigma, se mostrou uma boa alternativa, mas possui um custo muito alto e então também foi eliminado.

Dos sensores restantes, o sensor FHT20 do fabricante Liohm, foi a melhor alternativa, apesar de ter um preço mais alto e ser um pouco menos preciso (entretanto está compatível com os requisitos do processo). Este sensor possui uma vantagem sobre os demais, pois já possui uma interface de saída compatível com controladores industriais (os demais sensores dependem de uma eletrônica auxiliar), ele se mostrou um sensor estável, e esses fatores compensam custo mais alto.

Entretanto por fatores de custo, o estudo de caso utilizou o AM2315 da empresa AOSONG (que é uma alternativa similar, mas com algumas limitações na interface de comunicação), pois já havia disponível duas unidades, e os testes do

funcionamento do sistema podem ser feitos utilizando este sensor sem comprometer a análise dos resultados.

Este sensor é baseado no CI DS18B20, montado em um módulo com um microcontrolador que gerencia a leitura das medidas do chip e disponibiliza essas medidas em uma *interface* I2C. Este sensor também vem encapsulado com um suporte de montagem em parede e elementos de fixação, e filtros para evitar acúmulo de poeira, viabilizando a instalação em ambientes externos.

As características deste sensor são descritas na sequência:

- tensão de operação: 3,3V a 5V;
- corrente máxima: 10mA;
- intervalo de umidade relativa: 0% ~100%;
- intervalo de temperatura:-20°C ~ 80°C;
- precisão: $\pm 2\%$ para umidade relativa, $\pm 0,1^\circ\text{C}$ para temperatura;
- suporte para montagem em parede ou inserção em tubos;

Uma característica deste sensor é a tensão de operação, que é compatível com a *SBC Raspberry Pi 3*, especificada no projeto facilitando sua integração para testes de bancada. Esse instrumento é fornecido com suporte de fixação e possui um encapsulamento, que viabiliza sua montagem em campo. A umidade relativa do processo está entre 85% ~ 95%, valor muito alto e que boa parte dos sensores comerciais não atende. Possui uma boa precisão da medição e tem um ótimo custo benefício.

Algumas desvantagens deste sensor são:

- Preço alto (entretanto possui um bom custo benefício);
- Interface de comunicação baseada em I2C, isso limita o comprimento de seu cabo para aproximadamente 3m, forçando a instalação de um conversor de sinal em campo, próximo ao ponto de instalação do instrumento, para converter em outro protocolo de comunicação (ex: RS485, Hart, Ethernet), que possa transmitir essa informação para distâncias maiores;
- Outro fator desvantajoso, é que apesar de possuir uma interface de comunicação em I2C, não é possível mudar o endereço do dispositivo, o que inviabiliza o uso de mais de um sensor em barramento I2C.

Com exceção do alto custo, essas desvantagens não são observadas no sensor FHT20, conforme mencionado na análise comparativa, e esse é o principal diferencial do FHT20 sobre o AM2315.

A figura 13 mostra o sensor de temperatura e umidade especificado para este projeto.

Figura 13 – Sensor de Temperatura e Umidade AM2353



Fonte: LIOHM,2019

3.3.4. MEDIDOR DE CONSUMO DE ENERGIA

Para este projeto o medidor de consumo de energia elétrica, deverá fazer a medição da potência consumida por hora, e fornecer uma interface de comunicação que possa ser integrada com a SBC e sua instalação deverá ser em trilho DIN 35mm para que o mesmo possa ser montado no interior do painel de automação.

Existem diversos medidores de consumo que atendem estas características, e possuem uma interface de saída que variam desde uma saída pulsada até uma saída serial RS485 Mod Bus. Também podem fazer a medida de tensão, corrente, potência ativa, potência reativa, fator de potência, frequência.

Para este projeto foi adquirido um modelo simples e com um custo baixo, que faz somente a medida do consumo acumulado, e possui uma saída pulsada. Os modelos que disponibilizam mais variáveis possuem um custo elevado e sua interface é feita em RS485 o que dificulta a integração com a SBC. Em um primeiro momento do projeto, somente a medida do consumo será monitorada, para contabilizar os gastos com a cadeia de produção, mas no futuro poderá ser integrada as demais variáveis para monitorar também a qualidade da energia elétrica usada no processo.

Algumas características do medidor de consumo adquirido para o projeto estão descritas abaixo:

Modelo: DDS238-1;
Tensão: 120V;
Corrente máxima: 32A;
Frequência: 50Hz / 60Hz;
Consumo: < 2W;
Interface: Saída pulsada (18 ~ 27V), 30ms.

A figura 14 mostra o medidor de consumo usado neste projeto.

Figura 14 – Medidor de Consumo de Energia



Fonte: DOMOTICX,2019

3.3.5. SENSOR DE CO₂

Os sensores de CO₂ instalados nas salas para cultivo de cogumelos, têm a função de realizar medições que podem servir de base para o controle do sistema de ventilação e para o monitoramento do nível de CO₂ no interior da sala.

Como existem diversos tipos de sensores para medição do nível de CO₂, essa análise comparativa tem o objetivo de selecionar qual é o tipo de sensor mais adequado para esta aplicação.

A tabela 2, mostra os tipos de sensores envolvidos nesta análise, bem como as características consideradas. É desejável selecionar um sensor que no mínimo funcione em um intervalo de umidade de 85% até 100%, intervalo de nível de CO₂ de 300ppm até 5000ppm, tenha uma precisão compatível com o processo (+- 100ppm).

Tabela 2 – Tipos de sensores de temperatura e umidade

Instrumento	Intervalo Nível de CO ₂ [ppm]	Precisão	Estabilidade [ppm/Ano]
E+E Elektronik EE800	0ppm a 5000ppm	±50ppm	5 anos
E+E Elektronik EE871	0ppm a 5000ppm 0ppm a 10000ppm	±50ppm ±100ppm	?
Bravo Controls CO2-W-X	0ppm a 3000ppm	±70ppm	±50ppm
GE T8013	600ppm a 2000ppm	?	±20ppm

Instrumento	Umidade Ambiente	Interface de Saída	Custo
E+E Elektronik EE800	10% a 90%	RS485 – Modbus 4-20mA / 0-10V	?
E+E Elektronik EE871	0% a 100%	RS485 – Modbus	R\$ 5.707,00 ²
Bravo Controls CO2-N-LCD	0% a 95%	RS485 – Modbus 4-20mA / 0-10V	USD 72,00 ¹
GE T8013	0% a 95%	Relé	R\$ 803,36 ¹

Fonte: 2019, Autor

Notas:

1. Esse valor refere-se ao intervalo de 20% a 80%. Essa precisão não é constante e depende do intervalo de operação;
2. Este sensor também possui a medição de temperatura e umidade;

De acordo com a análise, o melhor sensor do ponto de vista técnico é o E+E Elektronik EE871, entretanto ele foi desconsiderado do projeto devido ao seu alto custo, apesar de possuir um sensor de temperatura e umidade junto com a medição do nível de CO₂.

O sensor GE T8013 apesar de ser uma boa alternativa, ele não possui uma saída analógica, ficando limitado a acionar e desacionar equipamentos, e desta forma esse sensor também foi desconsiderado.

O sensor E+E Elektronik EE800 possui uma limitação na umidade relativa do ambiente de operação (10% a 90%), o que acaba sendo um problema para a aplicação do projeto, e desta forma esse sensor também foi desqualificado.

O sensor escolhido para o projeto é o Bravo Controls CO2-N-LCD. Ele é a alternativa mais viável para este projeto (entre os instrumentos analisados), apresentando um custo mais acessível e uma boa especificação, atendendo aos requisitos do projeto. Apesar do intervalo de medição deste sensor não estar dentro do esperado (até 5000 ppm), ele será utilizado para manter o nível de CO₂ abaixo de 1000ppm, então isso não é um problema para o controle do sistema de ventilação. Já para o monitoramento, caso o nível de CO₂ esteja muito alto na fase inicial do cultivo, pode eventualmente haver uma perda de informação devido a esta limitação do sensor.

Este sensor foi projetado para ser montado na parede e possui uma saída que pode ser escolhida, entre: serial RS485 ModBus, tensão 0-5V e corrente 4-20mA.

A figura 15 mostra o medidor de CO₂ usado neste projeto.

Figura 15 – Sensor de CO₂



Fonte: BRAVO CONTROLS, 2019

3.4. CONTROLE DA TEMPERATURA

O ajuste da temperatura será feito por meio de dois atuadores um aquecedor e um refrigerador. Não é recomendado o uso de ar condicionado residencial para este tipo de aplicação, pois este tipo de equipamento não foi projetado para funcionar ininterruptamente (não necessariamente na sua potência nominal), seu rendimento é muito baixo em ambientes de alta umidade, além de ressecar o ar durante seu funcionamento, o que é prejudicial para o processo.

As salas para cultivo de cogumelos necessitam de um aquecedor para aumentar a temperatura interna para compensar a perda de calor para o ambiente em dias frios. Como existem diversos tipos de aquecedores, essa análise comparativa tem o objetivo de selecionar qual é o tipo de aquecedor mais recomendado para esta aplicação.

A tabela 3, mostra os tipos de aquecedores envolvidos nesta análise, bem como as características consideradas. É desejável selecionar um aquecedor que seja seguro, evite o ressecamento do ar e tenha um custo baixo.

Tabela 3 – Tipos de aquecedores e características

Tipo de Aquecedor	Custo Médio ¹	Resseca o ar do ambiente	Seguro para ambientes úmidos/fechados
Elétrico/Óleo	R\$240,00	Não	Não ²
Termoventilador	R\$89,50	Sim	Sim
Calefator de parede	R\$1265,00	Não	Sim
Elétrico/Cerâmico	R\$131,90	Pouco	Não ²
Climatizador Aquecedor	R\$589,30	Não	Sim
Aquecedor a gás	??	Não	Não ³
Condicionador de ar	R\$1103,60	Sim	Sim

Fonte: 2019, Autor

Notas:

1. Com exceção do Condicionador de Ar (2500W) e do Aquecedor a gás, todos os equipamentos foram considerados uma potência de aquecimento de 1500W;
2. Não recomendado para ambientes úmidos por risco de choque elétrico;
3. Não recomendado para ambientes fechados por risco de vazamento de gás;

De acordo com a análise, pode-se eliminar os aquecedores a gás por não serem uma alternativa segura em ambientes fechados. Pode haver vazamento de gás no interior da sala de cultivo podendo causar acidentes com vítimas fatais.

Os condicionadores de ar e os termos ventiladores podem ser eliminados por ressecarem demais o ar do ambiente, o que é prejudicial para o processo de cultivo dos cogumelos.

Os aquecedores elétricos a óleo e cerâmico não têm seu uso recomendado para ambientes úmidos devido à riscos de choque elétrico, entretanto se medidas de proteção forem tomadas como: energização por disjuntor diferencial residual, aterramento do equipamento, limitação ao acesso do equipamento e manuseio permitido somente com o equipamento desenergizado, pode-se considerar o uso destes equipamentos como uma alternativa viável.

Os climatizadores de ar no modo aquecedor e os calefatos elétricos são alternativas viáveis, mas são mais caras se comparados ao aquecedor cerâmicos e a óleo. Uma vantagem é que o climatizador pode ser utilizado também como umidificador, mas não como resfriador, pois a eficiência de seu funcionamento só é alta para ambientes muito secos, o que não é o caso da aplicação em questão.

Deste modo o aquecedor recomendado para o projeto é o aquecedor a óleo, pois esse tipo de aquecedor não causa ressecamento do ar no ambiente e são uma alternativa barata. Apesar de existir um certo risco relacionado à choques elétricos em ambientes úmidos, é possível mitigar isso com medidas de segurança como:

alimentação por disjuntores residuais diferenciais DRs, aterramento do equipamento, limitação ao acesso do equipamento e sempre manusear o equipamento com ele desenergizado.

Com relação ao arrefecimento da sala o equipamento mais recomendado é uma unidade condensadora + uma unidade evaporadora. Este equipamento é robusto e foi projetado para operar durante longos períodos de tempo de forma ininterrupta, além de ser possível enviar comandos elétricos. A figura 16 mostra uma unidade condensadora recomendada para esta aplicação.

Figura 16 – Unidade Condensadora



Fonte: MULTIFRIO, 2019

3.5. CONTROLE DA UMIDADE

O ajuste da umidade será feito por um sistema contendo uma bomba centrífuga, tubulações de PVC, bicos nebulosos e válvulas do tipo “borboleta”.

A água quando submetida a alta pressão e ejetada por um orifício pequeno é aspergida, fazendo com que sua absorção pelo ar aconteça com mais facilidade. Essa tubulação deve ser instalada no teto, fazendo com que a água seja aspergida no ar forçando o aumento da umidade ambiente.

Para etapa futura, válvulas solenóides serão instaladas nesse sistema e assim adicionando a possibilidade de utilizar a mesma bomba centrífuga para efetuar a irrigação dos cogumelos de forma automatizada.

3.6. CONTROLE DO NÍVEL DE CO₂ E SISTEMA DE VENTILAÇÃO

O ajuste do nível de CO₂ será feito diretamente pelo sistema de ventilação, que têm a função de renovar o ar do interior da estufa de cultivo. Os ventiladores especificados para esta aplicação são os ventiladores axiais para instalação em dutos. Os ventiladores mencionados neste projeto são mostrados na figura 17:

Figura 17 – Ventilador Axial



Fonte: GARDEN HIGH PRO, 2019

O sistema de ventilação também é utilizado para ajustar a temperatura interna da estufa quando há uma grande diferença de temperatura no interior e exterior da estufa. Nas situações em que a temperatura interna é baixa e a temperatura externa é alta e se deseja aumentar a temperatura, acionar os ventiladores pode ser uma opção com menos gasto de energia elétrica se comparada a um aquecedor. Essa mesma estratégia também pode ser aplicada quando a temperatura interna é alta e a temperatura externa é baixa e se deseja diminuir a temperatura interna da estufa de cultivo.

3.7. SERVIDOR E SISTEMA SUPERVISÓRIO

Uma etapa importante deste projeto é criar uma interface de comunicação entre o produtor de cogumelos e uma estufa de cultivo. Um pré-requisito para esta solução, é criar algo que seja fácil de usar. Com base nisso foi pensado e usar uma página HTML para este fim, já que hoje em dia quase todo mundo faz o uso da *internet* e as pessoas já estão acostumadas a interagir com informações apresentadas em páginas HTML.

Entretanto, as páginas HTML surgiram com um propósito de serem estáticas, ou seja, elas possuem informações que são apresentadas, mas que não podem ser modificados, pois o HTML não é uma linguagem de programação e sim uma

linguagem de marcação. Como é interesse dar uma dinâmica maior para a *interface* a ser desenvolvida, foi escolhida a linguagem de programação *Javascript* para esta finalidade, que pode ser vinculada à uma página HTML.

A linguagem *Javascript* foi usada para criar gráficos dinâmicos, usando o *framework* D3.js (<https://d3js.org/>). Essa é uma poderosa ferramenta para criação de gráficos dinâmicos onde é possível interagir com eles, alterando a escala, aplicando zoom, exibir ou ocultar informações pertencentes aos gráficos, etc.

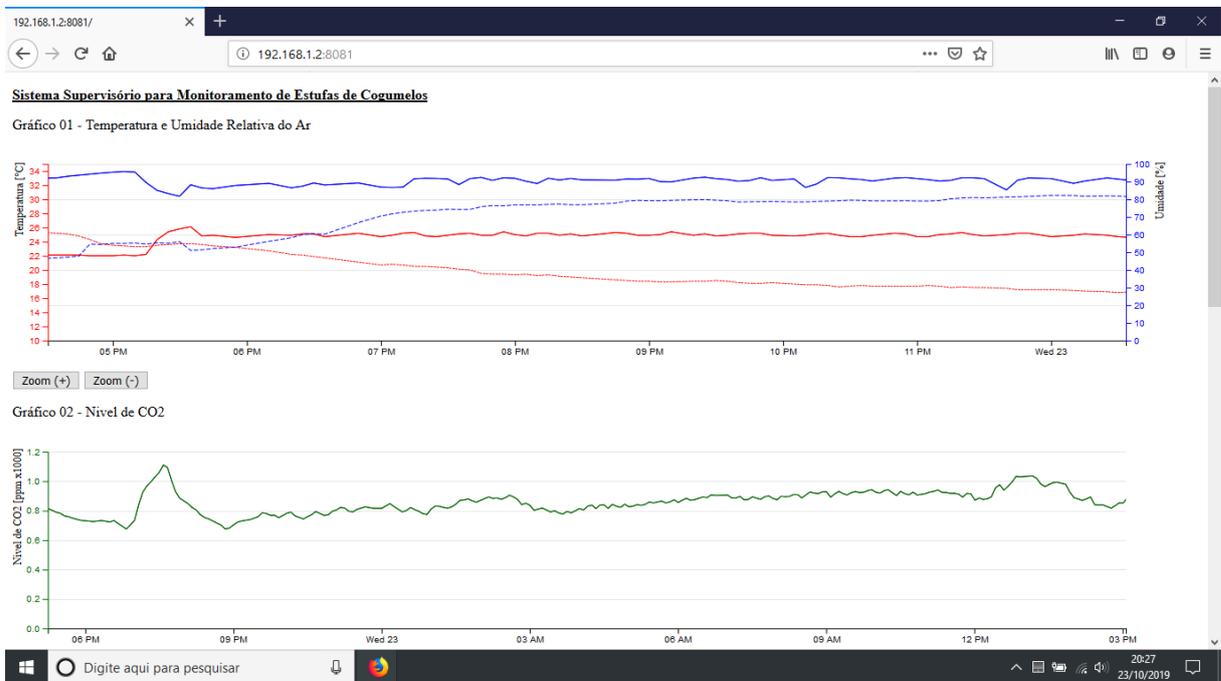
A biblioteca D3.js, pode apresentar dados de arquivos como CSV ou JSON, que serão realimentados pelo servidor para que os dados exibidos sejam atualizados com o passar do tempo. Foi escolhido o arquivo no formato CSV ao invés do JSON, pois o CSV possui menos redundância nas informações o que reflete em um menor consumo de memória para armazenar os dados comparado ao JSON. A desvantagem do CSV é que não há a possibilidade de criar hierarquias para uma melhor organização das informações armazenadas. Entretanto essa limitação não afeta em nada a aplicação dele neste projeto.

Usando um roteador foi instalada uma rede ethernet *wireless* com o sinal tendo alcance suficiente para chegar na sala de cultivo onde está instalada SBC. Esta eletrônica executa um aplicativo servidor, feito em Node, que ao receber uma requisição HTTP ela responde com uma página HTML, que contém todas as informações necessárias referentes ao processo de cultivo.

Este servidor possui um recurso de roteamento, permitindo que sejam enviadas informações do cliente para o servidor, utilizado para efetuar ações como: ajuste de *set-points* e acionamento de equipamentos. Com o Ajax o cliente consegue reenviar uma solicitação para o servidor para atualizar uma informação específica da página (não toda a página), fazendo com que não seja necessário recarregar toda a página para acrescentar somente a última leitura feita pelos sensores. Este servidor também tem acesso ao banco de dados onde consegue realizar escritas e leituras nas tabelas.

O código escrito do servidor é mostrado no apêndice B “Código do servidor *node_server.js*” e o código da página HTML é mostrado no apêndice C “Código da página HTML *index.html*”. A figura 18 mostra a tela do sistema supervisorio.

Figura 18 – Sistema supervisorio

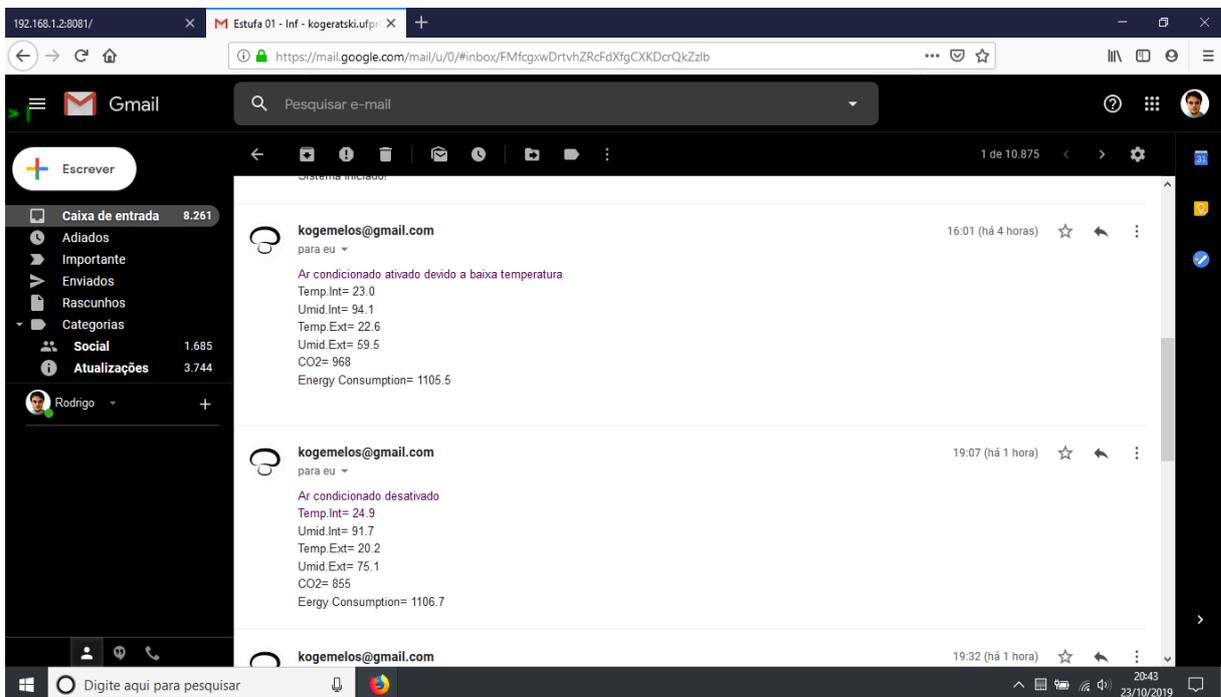


Fonte: 2019, Autor

Sempre que ocorre uma ação de acionar ou desacionar equipamentos ou quando o sistema é reiniciado, o mesmo envia um e-mail informando qual foi o motivo que levou a tal ação, bem como algumas informações adicionais.

A figura 19 mostra um exemplo de *e-mail* enviado pela sala de cultivo.

Figura 19 – E-mail enviado pela sala de cultivo



Fonte: 2019, Autor

Com exceção do envio de *e-mails*, todos os recursos mencionados neste capítulo podem ser configurados para funcionar em uma rede local, sem depender

de um acesso à *internet*, que era a proposta original do projeto. A possibilidade de implementar o recurso de envio de *e-mail* foi avaliada durante o desenvolvimento, e por ser um recurso muito útil, decidiu-se implementá-lo como um recurso adicional.

3.8. SOFTWARE PRINCIPAL

Para este projeto foi escolhida a linguagem de programação Python para o desenvolvimento das aplicações embarcadas. A escolha desta linguagem se deu pelo fato do fabricante do *Raspberry Pi* recomendar seu uso para o desenvolvimento das aplicações embarcadas. Desta forma, a maior parte do material de apoio para a implementação dos projetos usando o *Raspberry Pi* está nesta linguagem.

Outros fatores que contribuíram para a escolha da linguagem *Python*, é que a mesma é fácil de aprender e é gratuita. O software responsável pela leitura dos sensores e acionamento dos equipamentos é mostrado no 7.4 “Código do software main.py”.

Este software basicamente faz a leitura dos sensores (uma leitura a cada 5 minutos), armazena estas informações no banco de dados, realiza o acionamento dos equipamentos, envia *e-mail* com informações e eventos importantes do processo de cultivo.

3.9. BANCO DE DADOS

Já foi mencionado anteriormente, que existe a necessidade de armazenar informações na memória local do servidor, ou em qualquer outra mídia eletrônica, como um *pen-drive*. Atualmente a forma mais empregada para armazenar dados é usando os bancos de dados, que podem ser categorizados como relacionais e os não relacionais. No capítulo “2.6 Banco de Dados”, já foi feita uma abordagem sobre o assunto e explicado qual a diferença entre esses dois tipos de banco de dados. Neste capítulo será discutido qual tipo de banco de dados será implementado no projeto, qual a interface usada para criar o banco e a topologia proposta para o projeto em questão.

3.9.1. BANCO DE DADOS RELACIONAL VS NÃO RELACIONAL

Após ser feita uma pesquisa sobre os bancos de dados, o próximo passo foi decidir qual o tipo de banco de dados será usado, se será usado um banco de dados relacional ou um banco não relacional. Inicialmente, foi levantado quais informações podem ser armazenadas no banco de dados a ser definido, não levando em

consideração o grau de importância/utilidade desta informação, mas sim o maior número de informações possíveis que podem ser armazenadas. A intenção é fazer o dimensionamento do banco de dados, considerando um pior caso (muitas informações armazenadas), para que na fase de projeto possa ser avaliado como será o crescimento dos dados ao longo do tempo e se a SBC em questão vai conseguir manipular o volume de informações a serem armazenados.

Possivelmente esse banco de dados, independentemente do tipo escolhido, sofrerá adequações no futuro, caso esse projeto seja de fato implementado na prática. Se este for bem dimensionado, na medida que for necessário, mais informações podem ser acrescentadas no banco de dados sem problemas, como: limite de dados inseridos, lentidão no armazenamento e leitura das informações, falta de memória de armazenamento, etc.

Algumas das informações que podem ser levantadas estão descritas a seguir:

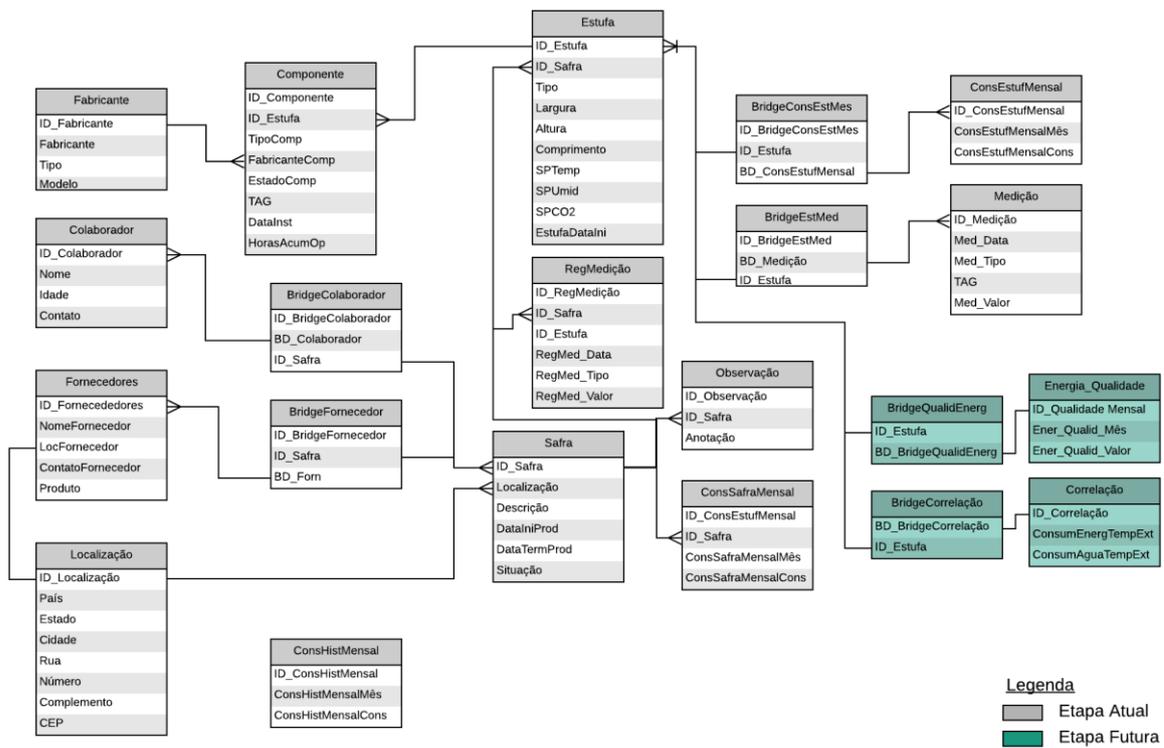
- Nome das safras (usada para distinta cada produção);
- Descrição das safras (informações adicionais para cada safra);
- Nome do produtor (pode ser que haja várias pessoas trabalhando na safra);
- Nome/Endereço/Contato dos fornecedores (rastreadabilidade dos insumos);
- Localização das estufas (as estufas não podem estar em vários locais);
- Método de esterilização (cadastrar para fazer comparações dos métodos);
- Situação atual da produção (produzindo, encerrada, contaminada, etc.);
- Tipo da estufa (câmara fria, contêiner, alvenaria);
- Dimensões da estufa (altura, largura, comprimento);
- Tipo de sistema de umidificação, ventilação ou aquecimento;
- Sensor de temperatura (marca, modelo, tipo de instalação);
- Sensor de umidade (marca, modelo, tipo de instalação);
- Sensor de CO₂ (marca, modelo, tipo de instalação);
- Medidor de energia elétrica (marca, modelo, tipo de instalação);
- Sensor de presença (marca, modelo, tipo de instalação);
- Hidrômetro (marca, modelo, tipo de instalação);
- Datas importantes referentes ao cultivo;
- Tempo de acesso ao interior da estufa (tempo médio, tempo acumulado);
- Temperatura média;
- Umidade média.

Ao levantar essas informações, percebe-se que elas são facilmente organizadas em grupos/tabelas, como por exemplo: Informações sobre as estufas, Informações sobre os fornecedores, informações sobre os equipamentos, etc. Desta forma foi escolhido usar um banco de dados relacional, que é o ideal para armazenar informações agrupadas em um conjunto de tabelas.

3.9.2. TOPOLOGIA DO BANCO DE DADOS

A proposta inicial de topologia do banco de dados relacional é mostrada na figura 42. Naturalmente que mais campos podem ser acrescentados no banco de dados, pois o mesmo foi desenvolvido usando a linguagem SQL. A figura 20 mostra as tabelas do banco de dados e a relação entre elas.

Figura 20 – Topologia do Banco de Dados



Fonte: 2019, Autor

3.9.3. BANCO DE DADOS EMBARCADO SQLITE

Uma vez definido o tipo de banco de dados e a topologia do mesmo, o próximo passo foi definir qual a solução a ser usada para criar o banco e fazer a interface de comunicação com o mesmo. O site <http://embedded-databases.com/> cita diversas opções de banco de dados embarcados, e é esperado que quase todos possam ser implementados para o projeto em questão. Pois o banco de dados a ser

desenvolvido não é tão complexo, não há imagens e arquivos armazenados no banco de dados e também é esperado que o intervalo de tempo entre as leituras dos sensores seja alto (minutos), pois a variação de umidade e temperatura por exemplo são muito lentas, o que reflete em um banco de dados com crescimento de informações lenta.

Entretanto, como critério de escolha foram considerados os seguintes fatores:

- Solução gratuita (iniciar com uma solução gratuita e evoluir para paga se houver necessidade no futuro);
- Programa leve (pois deve funcionar no *Raspberry Pi Model 3B*, que possui limitações de memória/processamento);
- Boa documentação de apoio e suporte técnico (como não tenho experiência com programação de banco de dados isso é um diferencial);
- Deve possuir uma interface com a linguagem de programação *Python*.

Com base nestes fatores, a solução escolhida foi SQLite (www.sqlite.org/). Pois ele é uma ferramenta *open source*, é leve, e é frequentemente mencionado nos fóruns online dedicados a projetos com o *Raspberry Pi*, onde o mesmo já foi testado com sucesso em diversas aplicações. Ele possui uma biblioteca para *Python* chamada “sqlite3”, que permite a interface de uma aplicação *Python* com o banco de dados. No site oficial do SQLite, está disponibilizado diversos documentos explicando como usar cada recurso do software o que facilita muito sua implementação.

3.10. USO DE CONTÊINERES COMO ESTUFAS DE CULTIVO

O projeto considera o uso de contêineres refrigerados do tipo “*Reefer HC 40 pés*”, adaptados como salas de cultivo de cogumelos.

A idéia do uso de contêineres refrigerados é dada pelos seguintes motivos:

- O contêiner refrigerado tem seu interior revestido de aço inox, que além de ser um material muito durável, ele não causa contaminações em alimentos;
- Esse tipo de contêiner conta com uma isolação de 100mm de poliuretano, o que é uma vantagem, já que evita a troca de calor com o ambiente e ajuda na economia de energia para o processo de cultivo;
- Um contêiner refrigerado do tipo HC, possui suas as seguintes medidas internas: 11,575m de comprimento, 2,29m de largura e 2,55m de altura. Que

podem comportar aproximadamente 2750kg de composto para cogumelos em seu interior;

- Os contêineres podem ser transportados. Assim pode-se criar uma nova sala de cultivo de cogumelos mais rapidamente quando comparado a uma nova estufa em alvenaria;

Foi realizado uma visita a uma empresa na região responsável por recondicionar e comercializar contêineres. Na figura 21 é mostrado o interior do contêiner mencionado neste capítulo, enquanto a figura 22 mostra um corte com o material isolante.

Figura 21 – Interior contêiner *Reefer HC 40* pés



Fonte: 2019, Autor

Figura 22 – Detalhe do material isolante



Fonte: 2019, Autor

O projeto tem o objetivo de propor uma solução completa para as estufas de cultivo, e usar um contêiner também cria a possibilidade dessa sala de cultivo ser alugada ao invés de ser vendida, já que a mesma pode ser transportada. O aluguel das salas de cultivo podem ser uma alternativa interessante para os produtores, já que isso reduz o investimento inicial para criar as estufas de cultivo, e também facilita o acesso à recursos tecnológicos que podem aumentar sua produção.

O uso desses contêineres como salas de cultivo, também possui algumas desvantagens, algumas delas são listadas a seguir:

- A largura interna é muito pequena para uma sala de cultivo de cogumelos, e isso pode dificultar a passagem de ar fresco entre as prateleiras. É necessário o uso de dutos para forçar a passagem do ar entre as prateleiras;
- A máquina que faz o controle de temperatura interna desse tipo de contêiner não pode ser utilizada para o controle de temperatura das salas de cultivo de cogumelos. Retirar essa máquina tem um custo e pode comprometer a isolamento térmica quando não realizado com qualidade. Se essa máquina for mantida em seu interior, ela acaba ocupando uma parcela do espaço interno, reduzindo a área útil de cultivo;
- O ideal é que haja uma antecâmara que dá acesso ao interior da sala de cultivo. Com o uso dos contêineres essa antecâmara não existe, deverá ser construída pelo lado de fora, ou deverá ser feita internamente, reduzindo a área útil de cultivo.

3.11. ANÁLISE DE VIABILIDADE ECONÔMICA

3.11.1. HIPÓTESE ESTUFA COM POUCO RECURSO TECNOLÓGICO

Neste capítulo é discutido uma análise de viabilidade econômica para uma empresa que aluga contêineres adaptados como salas de cultivos para cogumelos.

Na tabela 4 é mostrada a lucratividade anual de 4 salas de cultivo de cogumelos, com a mesma capacidade de carga de um contêiner (cada sala).

Tabela 4 – Lucratividade média obtida com cultivo de cogumelo paris in-natura, 4 estufas simples

Parâmetros		Observações
Dados Gerais		
Preço de Venda [R\$/kg]:	R\$15,00	Ref. CEASA Curitiba;
Número de fluxos/safra:	3	
Safra/Ano:	5	
Lucro Bruto/ Safra		
Quantidade de Composto [kg/safra]:	10000	Equivalente 4 contêineres;
Rendimento [%]:	25%	Valores entre 20% a 35%;
Produção de Cogumelos [kg/safra]:	2500	Peso in-natura;
Total [R\$]	R\$37500,00	
Despesas/Safra		
Logística p/ compra de comp. [R\$]:	R\$514,69	
Logística p/ venda de cogumelo [R\$]:	R\$400,00	
Preço do Composto [R\$/ton]:	R\$993,00	
Consumo de Energia [kWh]:	2400	Consid. 600kWh p/ estufa
Custo Energia [R\$/kWh]:	R\$0,79	
Custo total composto [R\$]:	R\$9.930,00	
Gasto com Energia [R\$]:	R\$1896,00	
Terra de cobertura [R\$]:	R\$3200,00	
Manutenção [R\$]:	R\$208,00	
Embalagem [R\$]:	R\$2000,00	Consid. Band. 250g a R\$0,2
Demais Gastos (Água) [R\$]:	R\$606,67	
Total [R\$]	R\$18.755,36	
Lucratividade e Mão de Obra		
Lucro s/ mão de obra	R\$18.744,64	
Necessidade de M.O. [horas./safra]:	600	
Custo de M.O. c/ encargos[R\$/hora]:	R\$15,40	Salário mínimo PR;
Duração da safra [dias]:	R\$65,00	
Custo Total M.O. [R\$/safra]:	R\$9.240,00	
Custo de Produção p/ kg	R\$11,20	
Lucro c/ mão de obra	R\$9.504,64	
Lucratividade c/ mão de obra:	25%	
Lucro Líquido Anual		
Lucro por safra:	R\$9.504,64	
Lucro Líquido anual:	R\$47.523,22	
Lucro Líquido mensal:	R\$3.960,27	

Fonte: 2019, Autor

3.11.2. HIPÓTESE ESTUFA COM RECURSOS TECNOLÓGICOS

Se os recursos tecnológicos implantados nos contêineres conseguirem aumentar o rendimento de um pequeno produtor (de 25% para 30%), cada contêiner iria gerar uma receita de R\$17.750,00 por ano. Estes dados são demonstrados na tabela 5.

Tabela 5 – Lucratividade média obtida com cultivo de cogumelo paris in-natura, 4 estufas otimizadas

Parâmetros		Observações
Dados Gerais		
Preço de Venda [R\$/kg]:	R\$15,00	Ref. CEASA Curitiba;
Número de fluxos/safra:	3	
Safra/Ano:	5	
Lucro Bruto/ Safra		
Quantidade de Composto [kg/safra]:	10000	Equivalente 4 contêineres;
Rendimento [%]:	30%	Valores entre 20% a 35%;
Produção de Cogumelos [kg/safra]:	3000	Peso in-natura;
Total [R\$]	R\$45.000,00	
Despesas/Safra		
Logística p/ compra de comp. [R\$]:	R\$514,69	
Logística p/ venda de cogumelo [R\$]:	R\$400,00	
Preço do Composto [R\$/ton]:	R\$993,00	
Consumo de Energia [kWh]:	2400	Consid. 600kWh p/ estufa
Custo Energia [R\$/kWh]:	R\$0,79	
Custo total composto [R\$]:	R\$9.930,00	
Gasto com Energia [R\$]:	R\$1896,00	
Terra de cobertura [R\$]:	R\$3200,00	
Manutenção [R\$]:	R\$208,00	
Embalagem [R\$]:	R\$2400,00	Consid. Band. 250g a R\$0,2
Demais Gastos (Água) [R\$]:	R\$606,67	
Total [R\$]	R\$19.155,36	
Lucratividade e Mão de Obra		
Lucro s/ mão de obra	R\$25.844,64	
Necessidade de M.O. [horas./safra]:	600	
Custo de M.O. c/ encargos[R\$/hora]:	R\$15,40	Salário mínimo PR;
Duração da safra [dias]:	R\$65,00	
Custo Total M.O. [R\$/safra]:	R\$9.240,00	
Custo de Produção p/ kg	R\$9,47	
Lucro c/ mão de obra	R\$16.604,64	
Lucratividade c/ mão de obra:	37%	
Lucro Líquido Anual		
Lucro por safra:	R\$16.604,64	
Lucro Líquido anual:	R\$83.023,22	
Lucro Líquido mensal:	R\$6.918,60	

Fonte: 2019, Autor

Nesta análise foi considerado um rendimento médio de 25 para uma sala de cultivo simples. Mas usando recursos tecnológicos é possível aumentar a

lucratividade, pois é esperado um rendimento maior e uma necessidade menor de mão de obra.

Considerando 4 estufas com capacidade total de 10 toneladas de composto (4 contêineres) e rendimento de 25% o lucro líquido anual ficou em R\$47.523,22, mas se aumentarmos esse valor para 30% usando automação, o lucro líquido anual vai para R\$83.023,22, e cada contêiner iria gerar uma receita de R\$17.750,00 a mais do que o cenário anterior.

3.11.3. INVESTIMENTO E PAYBACK

O custo para a construção de uma estufa para cultivo de cogumelos reutilizando um contêiner refrigerado está próximo de R\$40.000,00, conforme detalhado na tabela 6.

Tabela 6 – Custo estimado para adaptar contêiner refrigerado

Material/Equipamento	Quantidade	Custo
Contêiner <i>Reefer HC</i> 40 pés	1	R\$12.500,00
Unidade Condensadora	1	R\$5.000,00
Unidade Evaporadora	1	R\$1.000,00
Prateleiras	2	R\$10.000,00
Painel de Automação (materiais)	1	R\$2.000,00
Instrumentos	4	R\$1.000,00
Diversos (mão de obra, materiais, etc.)	1	R\$8.500,00
Total Estimado		R\$40.000,00

Fonte: 2019, Autor

Considerando um investimento de R\$160.000,00 com uma taxa de juros de 10% a.a. e uma produção com 4 estufas, temos os seguintes cenários mostrados na tabela 7 (sendo 20% um cenário pessimista 25% e 30% otimista):

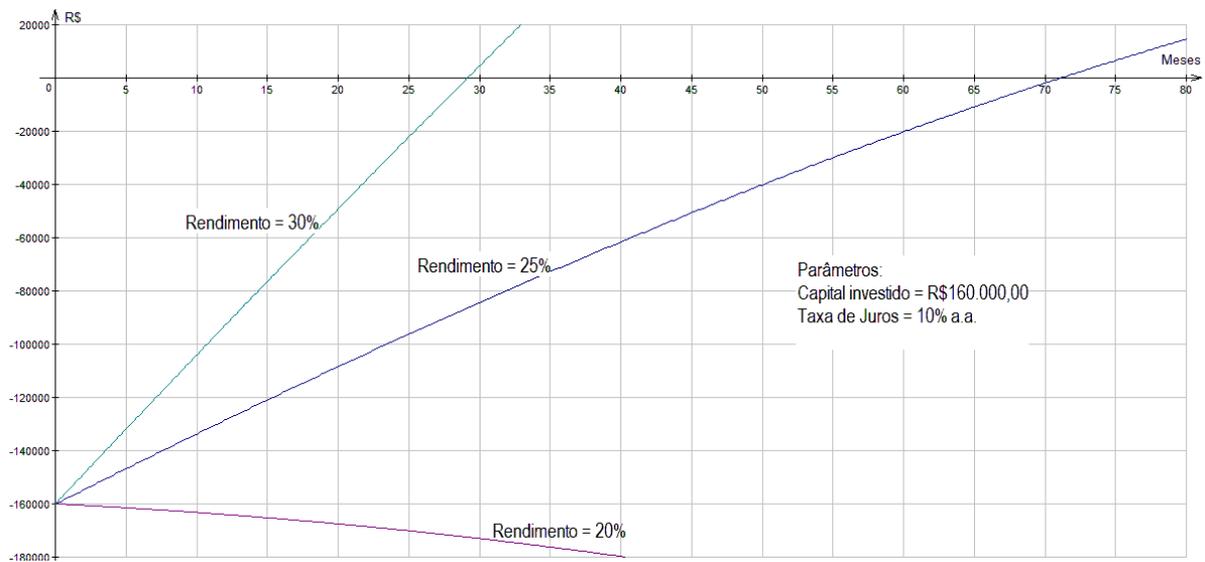
Tabela 7 – Investimento inicial e *Payback*

Investimento Inicial	Rendimento das estufas	Receita Mensal	<i>Payback</i> (meses)
R\$160.000,00	20%	R\$1001,93	∞
R\$160.000,00	25%	R\$3960,27	71,05
R\$160.000,00	30%	R\$6918,60	29,15

Fonte: 2019, Autor

A figura 23 mostra um gráfico com a evolução da receita obtida ao longo do tempo, com o cultivo e comercialização de cogumelos.

Figura 23 – Receita obtida com 4 estufas de cultivo de cogumelos ao longo do tempo



Fonte: 2019, Autor

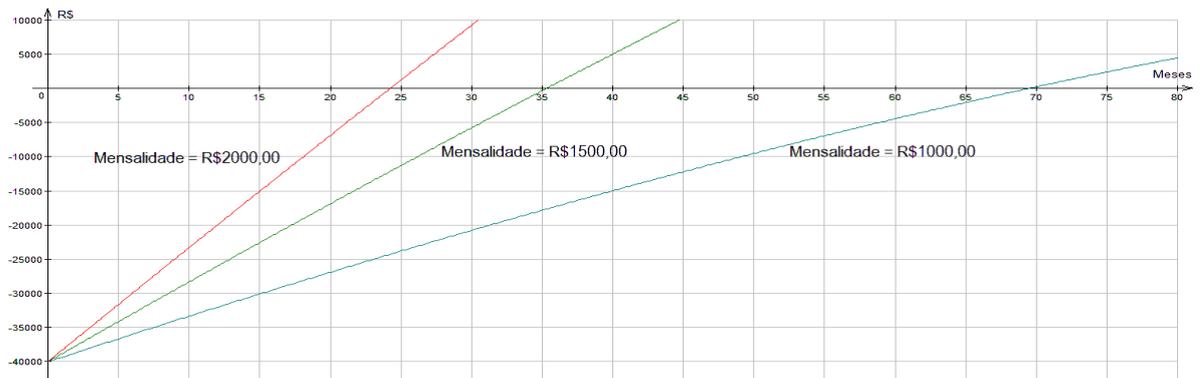
Para o caso de investir na montagem de um contêiner e alugar esta solução para um agroempreendedor, temos um cenário em que há um investimento de R\$40.000,00 para cada estufa e a mesma irá gerar uma receita mensal proveniente de um aluguel.

Conforme mostrado anteriormente na tabela 04, com cada estufa é possível gerar uma receita que pode variar de acordo com o rendimento obtido. Para que esse negócio seja viável ao agroempreendedor, o mesmo deve obter lucro mesmo pagando a mensalidade, que pode ser de R\$6918,60 se o mesmo conseguir um rendimento de 30% (caso otimista).

Logo a mensalidade cobrada teria que ser um valor abaixo de R\$6918,60 e quanto mais baixo for a mensalidade mais atrativo é para o cliente e conseqüentemente mais fácil de firmar um contrato. Entretanto há a necessidade de que haja um retorno financeiro, então a mensalidade não pode ser um valor muito baixo, pois isso iria inviabilizar o aluguel desta solução.

A figura 24 mostra alguns *paybacks* considerando os valores de mensalidades R\$2.000,00, R\$ 1.500,00 e R\$1.000,00. Os *paybacks* (em meses) obtidos foram de: 29,16, 71,05 e infinito, respectivamente.

Figura 24 – Payback para aluguel de estufas de cultivo de cogumelos



Fonte: 2019, Autor

3.12. PROJETO PAINEL DE AUTOMAÇÃO

O projeto do painel de automação foi desenvolvido com o software DraftSight 2018 x64 SP3 e é mostrado no apêndice G. Algumas etapas da montagem do painel são mostradas nas figuras 25 e 26, e a montagem final é mostrado na figura 27.

Figura 25 – Montagem do painel – corte dos trilhos



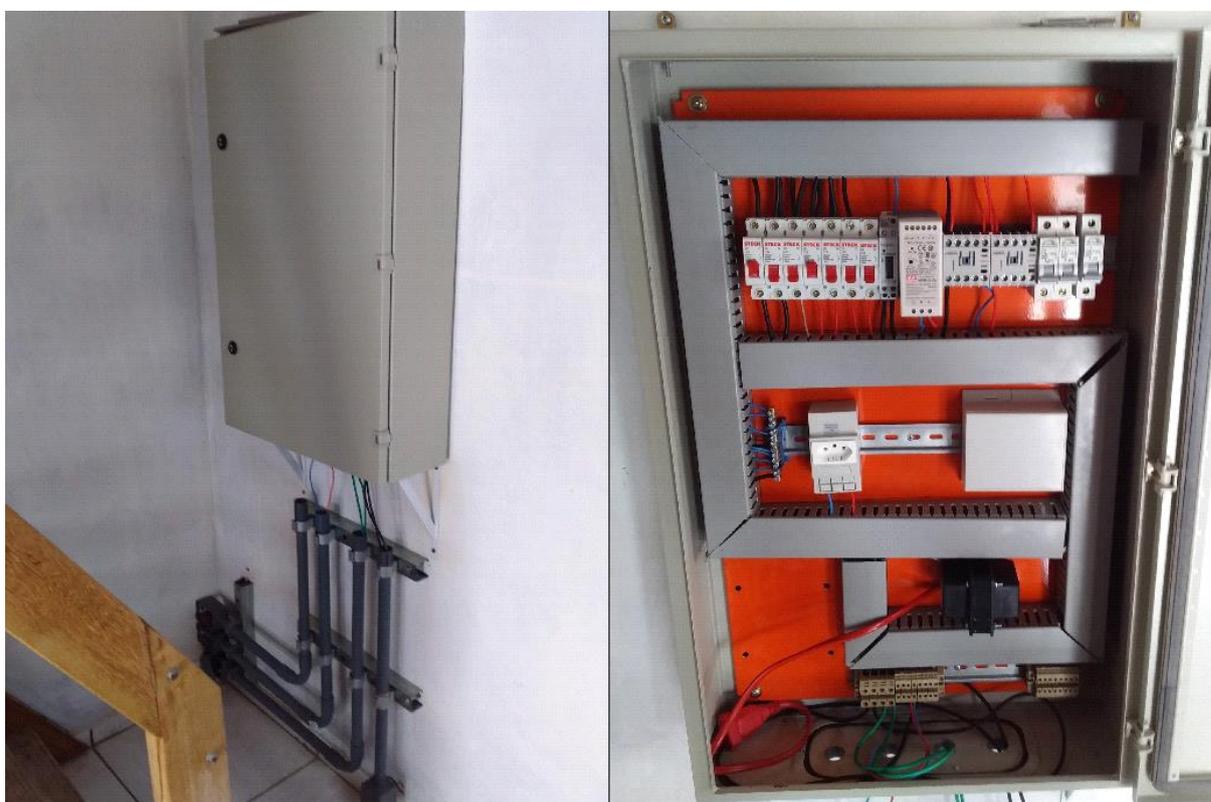
Fonte: 2019, Autor

Figura 26 – Montagem do painel – trilhos e canaletas



Fonte: 2019, Autor

Figura 27 – Montagem do painel – fixação na parede



Fonte: 2019, Autor

3.13. INFRAESTRUTURA DE MONTAGEM

Toda a infraestrutura elétrica para passagem dos cabos e alimentação dos equipamentos foi realizada durante o período de desenvolvimento deste projeto. Algumas etapas desta montagem são mostradas nas figuras 28, 29 e 30.

Figura 28 – Infraestrutura elétrica – instalação do sensor de CO₂



Fonte: 2019, Autor

Figura 29 – Infraestrutura elétrica – instalação das lumiárias



Fonte: 2019, Autor

Figura 30 – Infraestrutura elétrica – instalação dos eletrodutos



Fonte: 2019, Autor

3.14. ESTUDO DE CASO – ESTUFA DE CULTIVO

Para validar este trabalho foi elaborado um estudo de caso para testar a proposta apresentada pelo projeto em uma pequena estufa para cultivo de cogumelos. A estufa em questão fica localizada no distrito de Mariental (pertencente ao município da Lapa/PR) a aproximadamente 60 km de Curitiba/PR, ela possui

25m² de área, tem duas prateleiras que podem comportar até 108 sacos de composto (10 a 13kg), podendo gerar uma produção aproximada de 250kg a 350kg de cogumelo *in-natura* por cada safra (duração aproximada de 65 dias).

Esta estufa possui um ar condicionado split inverter 18.000btus (quente/frio) para fazer o controle da temperatura, um termômetro de álcool para aferir a temperatura interna e um sistema de ventilação improvisado com 8 *Coolers* 24Vcc. Também há um painel elétrico de distribuição que providencia os ramais de alimentação para os equipamentos e comporta os demais componentes, entre eles um medidor de consumo de energia elétrica (DDS 238-1).

A proposta do estudo de caso foi executar as seguintes ações:

- Realizar o projeto e montagem de um novo painel de automação;
- Substituir o quadro de distribuição atual pelo novo painel de automação;
- Instalar sensores eletrônico para fazer a medição da temperatura/umidade interna e externa da sala de cultivo;
- Instalar sensor eletrônico para medir o nível de CO₂;
- Substituir o sistema de ventilação pelo novo sistema proposto no projeto;
- Testar o sistema de supervisão proposto no projeto;
- Testar a SBC (*Raspberry Pi 3 Model B*) como eletrônica de controle.

Este estudo de caso não contempla alguns itens que foram propostos no projeto sob as seguintes justificativas:

- A utilização da unidade condensadora mais uma unidade evaporadora. Não foram contemplados no estudo de caso devido ao alto custo do equipamento. Desta forma foi decidido manter o sistema existente (ar condicionado);
- Não foi instalado um sistema de controle de umidade, por falta de tempo para efetuar as montagens necessárias, e também pelo alto custo dos equipamentos/materiais;
- Não foi testado os algoritmos de controle propostos no projeto, pois não houve meios de interligar o ar condicionado na SBC e controlar sua potência;

A montagem de toda a infraestrutura elétrica foi iniciada no início de setembro. A meta era finalizar toda a montagem antes do dia 26 de setembro de 2019, pois nesta data estava previsto a chegada do composto de cogumelos e logo seria dado início ao processo de cultivo.

A figura 31 e 32 mostram a chegada do composto para cogumelos no dia 26 de setembro de 2019, a figura 33 mostra as prateleiras com o composto acomodado.

Figura 31 – Transporte do composto para cogumelos



Fonte: 2019, Autor

Figura 32 – Detalhe da carga transporte do composto para cogumelos



Fonte: 2019, Autor

Figura 33 – Prateleiras com o composto de cogumelos



Fonte: 2019, Autor

Neste período foi desenvolvido um novo projeto do painel de automação que é mostrado no apêndice G, foi realizado a montagem deste painel, removido o painel de distribuição existente e instalado o novo painel de automação conforme já mostrado na figura 27 “Montagem do painel – fixação na parede”.

Também foi feita a montagem dos sensores de temperatura e umidade e do sensor de CO₂, conforme mostrados nas figuras 34 e 35.

Figura 34 – Instalação do sensor de temperatura e umidade externo



Fonte: 2019, Autor

Figura 35 – Instalação dos sensores internos



Fonte: 2019, Autor

O sistema de ventilação foi substituído pelo ventilador proposto no projeto (conforme mencionado no capítulo 4.6. “Controle do nível de CO₂ e sistema de ventilação”), instalado no interior de um tubo de PVC 100mm. A figura 36 mostra os ventiladores que foram adquiridos para a montagem do sistema de ventilação.

Figura 36 – Ventiladores tubular 100mm

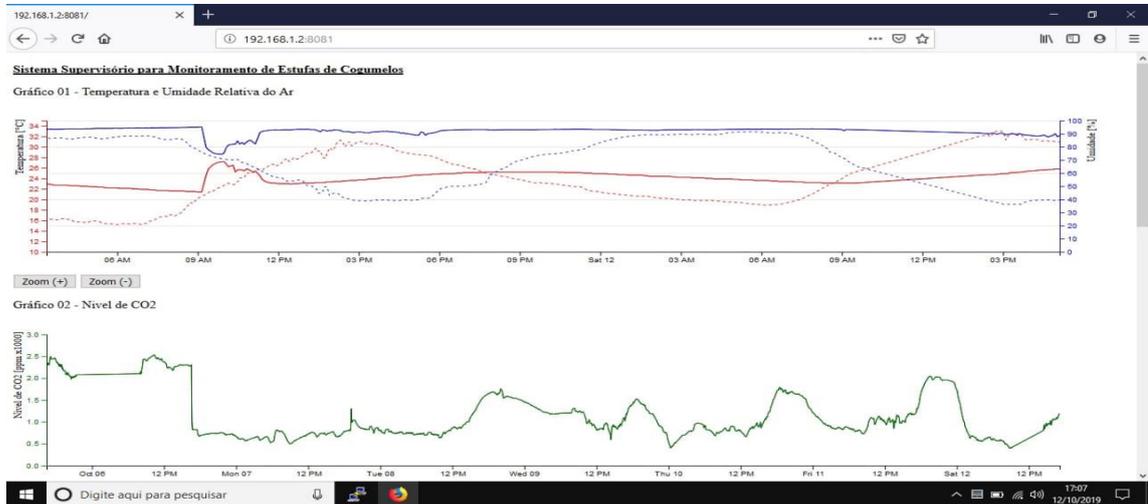


Fonte: 2019, Autor

Devido à carga de trabalho exigida para montar toda a infraestrutura, o sistema de monitoramento sofreu um atraso em sua implantação e teve seu funcionamento dado no dia 05 de outubro de 2019 (houve um atraso de 10 dias).

A figura 37 mostra o sistema de supervisão da estufa de cultivo.

Figura 37 – Sistema de supervisão para monitoramento da estufa



Fonte: 2019, Autor

Na figura 34 a linha pontilhada é referente as medidas externas (temperatura na cor vermelha e umidade na cor azul). A linha contínua é referente as medidas internas da estufa. Nos primeiros 25 dias é recomendado manter a temperatura interna em 25°C, entretanto é observado no gráfico que há muita oscilação pois ainda não há um sistema de controle efetivo implantado nesta sala de cultivo. Essa oscilação não é desejada e pode reduzir o rendimento do cultivo, mas não é suficiente para inviabilizar a produção.

Ainda na figura 34, o gráfico 2 mostra a variação de CO₂ no interior da sala. Neste momento ainda não está sendo realizada a troca de ar no interior da estufa, pois é desejável que o nível de CO₂ esteja alto nesta etapa (acima de 5.000ppm).

As variáveis de processo temperatura interna, temperatura externa, umidade interna e umidade externa, começaram a ser monitoradas a partir do dia 05. A leitura do nível de CO₂ foi integrada no sistema alguns dias depois e a integração do consumo de energia foi efetivada somente no final de outubro.

Os vídeos de apresentação e testes do sistema de monitoramento podem ser visualizados nos seguintes links:

<https://photos.app.goo.gl/UpkDdKdFmYAR117M9>

<https://photos.app.goo.gl/9YMuLYTSKGj59VPB6>

<https://photos.app.goo.gl/shgzMwJ4xmhuX4ok7>

4. RESULTADOS E DISCUSSÃO

Conforme mencionado nos objetivos específicos, a caracterização do processo de cultivo de cogumelos foi abordada nos capítulos 2.1. “Processo de Cultivo do *Agarius bisporus*” e no capítulo 4.2 “Avaliação do processo de cultivo”, as variáveis mais importantes (por ordem de prioridade) para esse cultivo são: a temperatura interna estufa, o nível de CO₂ e a umidade relativa do ar (interior da sala de cultivo). Outras variáveis também influenciam no processo, mas com menor importância.

O controle destas variáveis e os equipamentos envolvidos nisso foram abordados nos capítulos: 4.4. “Controle da Temperatura” que propôs um aquecedor a óleo como melhor alternativa para incremento da temperatura e uma unidade condensadora + evaporadora para arrefecimento do ambiente. Já o controle de umidade abordado no capítulo 4.5 “Controle da Umidade” propõe um sistema de incremento da umidade baseado em pressurização da água em tubos de PVC e bicos nebulizadores para aspergir vapor de água no interior da sala de cultivo, e o sistema de ventilação abordado no capítulo 4.6 “Controle do Nível de CO₂ e Sistema de Ventilação”, propõe o uso de ventilador/exaustor tubular em dutos de PVC para injetar/retirar o ar no interior da estufa.

O desenvolvimento do segundo objetivo específico mencionado no capítulo 1.1.2. “Objetivos Específicos”, está descrito no capítulo 4.7. “Servidor e Sistema Supervisório” e utilizou como base a pesquisa descrita nos capítulos 2.9. “Redes de Computadores”, 2.10. “Protocolo Ethernet” e também a topologia mostrada na figura 10 “Diagrama da Solução Proposta”. De uma forma geral foi programado um servidor em NODE.js (mostrado no apêndice B “Código do Servidor node_server.js”), embarcado na SBC e instalado em campo. Esse servidor pôde ser acessado por qualquer computador ou smartphone conectado na rede local conforme mostrado na figura 34 “Sistema de supervisão para monitoramento da estufa”.

O desenvolvimento do terceiro objetivo específico mencionado no capítulo 1.1.2. “Objetivos Específicos”, também está descrito no capítulo 4.7. “Servidor e Sistema Supervisório” e utilizou como base a pesquisa descrita no capítulo 2.8. “Linguagem de Programação Javascript”. O código dos recursos implementados é mostrado no apêndice C “Código da Página HTML index.html”. O funcionamento deste recurso pode ser visto no vídeo feito para apresentar o sistema de monitoramento proposto, mencionado mais adiante ainda neste capítulo.

Os demais softwares envolvidos no projeto estão descritos no capítulo 4.8. “Software Principal”, que usou como base a fundamentação teórica descrita no

capítulo 2.7. “Linguagem de Programação Python”, e este código é mostrado no apêndice A “Código do Software main.py”.

O quarto objetivo específico mencionado no capítulo 1.1.2. “Objetivos Específicos”, está descrito nos capítulos 4.3. “Seleção dos instrumentos e equipamentos”, que definiu o sensor de temperatura e umidade FHT20 do fabricante Liohm como melhor alternativa (dentre os candidatos considerados) e 4.9. “Banco de Dados” que determinou o banco de dados relacional como tipo de banco de dados mais recomendado e propõe o SQLite como interface recomendada para o projeto. Estes capítulos utilizaram como base os capítulos 2.1. “Medição de Temperatura”, 2.3. “Medição de Umidade Relativa do Ar” e 2.6. “Banco de Dados”.

A validação do projeto foi realizada com um estudo de caso (último objetivo específico mencionado no capítulo 1.1.2. “Objetivos Específicos”), e foi abordado no capítulo 4.14. “Estudo de Caso – Estufa de Cultivo”. O arquivo CSV com todas as medidas realizadas durante o estudo de caso pode ser acessado pelo seguinte link: https://drive.google.com/drive/folders/1NnF1RmUpfg8S_5VB8chsJgD4VZD_I9gO

4.1. OBSERVAÇÕES, PROBLEMAS E DIFICULDADES

Durante realização do estudo de caso foi possível validar o funcionamento de alguns componentes propostos, levantar dados e testar a proposta geral do projeto.

Com relação à instrumentação utilizada no estudo de caso para monitoramento da temperatura e umidade, foi observado algumas falhas de comunicação com o instrumento (tanto interno quanto externo a sala de cultivo). A causa desta falha foi a falta de conectores adequados para fazer as emendas do cabo de sinal.

Por falta de tempo e recursos foi utilizado um cabo de baixo custo com dois pares metálico e feito emendas com bornes e fita isolante, que em um primeiro momento funcionou de forma adequada, mas com o passar do tempo (aproximadamente 1 mês) começaram a aparecer as falhas, que foram sanadas refazendo as emendas. No final do estudo de caso não foi detectado nenhuma anormalidade com estes sensores (mesmo tendo funcionado durante 2 meses em ambiente muito úmido), e os mesmos continuaram realizando suas medidas de forma coerente não sendo observados problemas de desvio ou aumento da imprecisão. Deste modo pode-se concluir que o sensor de temperatura e umidade AM2315 pode ser utilizado para realizar o monitoramento da temperatura e umidade

das estufas de cultivo, desde que seja utilizado cabos e conectores de boa qualidade para evitar falhas de comunicação com o sistema de controle.

Com relação ao sensor de CO₂, este funcionou de forma ininterrupta sem apresentar nenhuma falha se mostrando uma alternativa confiável para o monitoramento do CO₂ na sala de cultivo. Entretanto foi observado um problema com o processo de cultivo, pois o sensor de CO₂ apesar de funcionar corretamente e servindo de base para o controle do nível desse gás no interior da estufa (foi mantido abaixo de 1000ppm), ainda assim foram observados que os primeiros cogumelos estavam nascendo deformados.

A principal causa deste problema é o excesso de CO₂, logo a hipótese adotada é de que as medidas realizadas pelo sensor instalado na parede não representam o nível de CO₂ próximo aos cogumelos, que foram inicialmente prejudicados. Por limitação do comprimento do cabo não puderam ser feitas medições mais próximas dos cogumelos, mas pôde-se concluir que o local da instalação escolhido é inadequado, e este deveria estar localizado em um ponto estratégico mais próximo aos cogumelos, para efetuar uma medição mais precisa. Após ser observado este problema o nível de ventilação foi aumentado e manteve o nível de CO₂ o mais baixo possível (muito próximo de 450ppm) e as deformações nos cogumelos não foram mais observadas.

Com relação ao medidor de consumo de energia, o modelo proposto no projeto é um modelo com comunicação serial *Modbus*. No estudo de caso foi utilizado um modelo muito similar (do mesmo fabricante), mas com saída pulsada. A safra foi iniciada no dia 26 de setembro de 2019 com um consumo de energia acumulado de 959,5kWh, e até o dia 22 de novembro de 2019 o consumo de energia acumulado estava em 1298,8kWh, ou seja 339,3kWh (lembrando que a safra ainda está em andamento).

Em relação à precisão do medidor o mesmo se mostrou satisfatório para medir e contabilizar o gasto com energia elétrica durante a safra. Por fatores de economia de recursos (o sensor proposto no projeto não é comercializado no Brasil e deve ser importado), o estudo de caso utilizou um medidor de consumo com saída pulsada (DDS238-1), que apresentou problemas de perda de sincronismo nas medições, quando a placa de controle era reiniciada, ou ficava inativa por longos períodos de tempo. Os pulsos gerados neste intervalo não eram contabilizados gerando uma diferença na medição realizada pelo sensor e o que o sistema contabilizou no banco de dados. Entretanto esse problema não é esperado para o

modelo proposto pelo projeto (DDS238-1 ZN), que possui comunicação serial, e tanto o valor instantâneo quanto o acumulado pode ser lido a qualquer momento.

A placa de controle (*Raspberry Pi Model 3B+*) apresentou muitos problemas durante o estudo de caso. O primeiro problema relatado é uma limitação da potência de recepção do sinal *wi-fi*, que ficou muito fraco dentro do painel de automação (mesmo este estando relativamente perto do roteador). Antes de instalar a placa foi feito testes com um *smartphone* posicionado dentro do gabinete metálico que captou o sinal *wi-fi* sem maiores dificuldades, já a SBC não obteve o mesmo desempenho. Para resolver este problema foi utilizada uma repetidora de sinal *wi-fi*, pois por limitações no local seria muito difícil levar um cabo de rede até a SBC.

O segundo problema observado com esta placa foi em relação à caixa plástica adquirida para instalação da mesma no trilho do painel de automação. Esta caixa não prevê a saída dos cabos dos pinos de sinal, sem que seja necessário cortar e furar a caixa (não foi encontrada nenhuma caixa comercial que não tivesse esse problema). Isso demanda tempo para efetuar a montagem e acaba baixa a qualidade da instalação, o que reflete em falhas de comunicação.

O sistema operacional embarcado desta placa foi corrompido ao longo dos testes, possivelmente ocasionado por oscilações na rede de alimentação, baixa qualidade no cartão SD, interferência eletromagnética ou defeito de fabricação. Isso causou uma parada total no sistema e perda de 1 semana de leituras dos sensores, também foi necessário mais de 24h para reinstalar e reconfigurar a placa para que a mesma voltasse a operar. Isso foi a segunda vez que aconteceu, sendo que a primeira vez que o cartão SD foi corrompido, aconteceu com a placa guardada e sem operação.

Essa SBC também apresenta perda de sincronismo no relógio quando opera por longos períodos de tempo sem estar conectada na *internet*. Para resolver este problema é necessário utilizar um circuito *Real Time Clock* externo, o que não é interessante. O objetivo do projeto era operar com este sistema sem depender do acesso à internet, operando somente em uma rede local.

O *software* projetado apresentou alguns erros de projeto, que foram detectados quando o sistema foi colocado em operação. Estes problemas foram sanados e então não foram mais observados erros ou falhas relacionadas ao código do *software* de controle.

O sistema supervisorio funcionou corretamente, mas com um tempo excessivo no carregamento da página. Essa demora no carregamento foi maior quando acessado pelo *smartphone*. As funcionalidades da página foram melhores

aproveitadas quando acessados pelo computador quando comparados pelo *smartphone*.

Também houve problemas de queda de energia durante o estudo de caso. No dia 16 de setembro, devido à um acidente que causou a derrubada de um poste da concessionária de energia local, a estufa ficou sem energia durante quase 12h. Nesta região também é muito comum acontecer apagões em dias de chuva, e há uma alta incidência de raios que causam oscilações na rede de energia elétrica.

4.2. OPORTUNIDADES DE MELHORIAS

Diante dos problemas expostos no capítulo 5.1. “Observações, Problemas e Dificuldades”, é sugerido algumas mudanças no projeto a fim de melhorar o mesmo:

- Melhorar a qualidade das instalações dos instrumentos, utilizando conectores adequados para realizar as conexões dos sensores e utilizar cabos próprios para sinal elétrico;
- Redefinir o ponto de instalação do sensor de CO₂, posicionando o mesmo em um local mais próximo dos cogumelos, com o objetivo de realizar uma medição mais confiável;
- Dar prioridade em fazer a conexão da SBC com a rede *ethernet* por meio de cabos ao invés da rede *wi-fi* a fim de aumentar a confiabilidade da comunicação;
- Rever a especificação da SBC e avaliar a possibilidade de utilizar uma placa eletrônica de maior qualidade ou utilizar um Controlador Lógico Industrial (CLP). Também é uma alternativa tomar medidas para que sejam mitigados os problemas observados neste estudo de caso, referentes à placa de controle;
- Utilizar um estabilizador de tensão para evitar problemas de oscilações na rede, o que pode causar falhas na placa de controle;
- Utilizar uma fonte UPS ou No-Break para que não falte energia no sistema de monitoramento em eventos de queda de energia;
- Otimizar o sistema supervisório para que seja reduzido o tempo de carregamento das páginas;
- Criar um aplicativo para *smartphone* próprio para acessar o sistema supervisório ou criar uma página simplificada que viabilize esse acesso de forma mais rápida por este tipo de dispositivo.

5. CONSIDERAÇÕES FINAIS E ESTUDOS FUTUROS

De uma forma geral ainda há muita coisa que pode ser trabalhada e melhorada neste projeto. Não foi possível testar tudo o que foi proposto, mas pôde ser gerado diversos resultados, que podem servir de base para estudos futuros.

5.1. ESTUDOS FUTUROS

O sistema de monitoramento não está completo, e ainda há mais variáveis de processo que podem ser integrados ao sistema. Ou seja, há oportunidades de continuar evoluindo este presente trabalho.

É possível desenvolver um estudo mais específico voltado à confiabilidade no uso das SBCs em projetos aplicados. Conforme discutido neste trabalho, o uso de uma SBC para uma aplicação ininterrupta não foi satisfatória conforme esperado.

Este trabalho pode ser utilizado como base para o desenvolvimento de um projeto de controle de uma estufa para cultivo de cogumelos.

O sistema de software pode ser otimizado, utilizando técnicas de testabilidade para detectar e corrigir falhas de projeto.

Existem muitas oportunidades de desenvolvimento de sensores, instrumentos eletrônicos ou equipamentos específicos para o cultivo deste alimento.

6. REFERÊNCIAS BIBLIOGRÁFICAS

ALBUQUERQUE, Pedro Urbano Braga de; THOMAZINI, Daniel, **Sensores Industriais – Fundamentos e Aplicações**, 4ª edição, Editora Érica, 2005

ARAÚJO, Wellington Junior. **Análise de Custo da Produção de Champignon: Estudo em uma Propriedade Rural no Município de Domingos Martins - ES**. Revista Científica Intelletto, Venda Nova do Imigrante , ES, v.1, n.1, 2016, p113.

BASSALO, José Maria Filardo. **Curiosidades da Física**. Disponível em:
< <http://www.seara.ufc.br/folclore/folclore481.pdf> />. Acesso em: 17 de abril 2018

BRAVO CONTROLS. **CO₂ Sensor**, 2019. Disponível em:
< <https://bravocontrols.com/shop/co2-sensor-2/> />. Acesso em: 20 de junho 2019

BSE PAINÉIS. **Produtos**, 2019. Disponível em:
< <https://bsepaineis.com.br/paineis/quadro-de-comando-ip66> />. Acesso em: 22 de fevereiro 2019

CIRCUIT GLOBE. **Ammeter Shunt**, 201-. Disponível em:
< <https://circuitglobe.com/ammeter-shunt.html> />. Acesso em: 15 de setembro 2018

DB-ENGINES. **DB-Engines Ranking**, 2019. Disponível em:
< <https://db-engines.com/> />. Acesso em: 15 de abril 2018

DOMOTICX. **Modbus – KWh meter – DDS238-1 ZN**, 2019. Disponível em:
< <http://domoticx.com/modbus-kwh-meter-dds238-1-zn/> />. Acesso em: 14 de março 2019

EIRA, Augusto Ferreira. **Cogumelos comestíveis, um mercado promissor**. Disponível em: < <https://www.cpt.com.br/artigos/cogumelos-comestiveis-um-mercado-promissor> />. Acesso em: 20 de março 2018

GARDEN HIGH PRO. **Profan Axial Inline Fan**, 2019. Disponível em: < <http://www.gardenhighpro.com/> />. Acesso em: 18 de março 2019

GOLDSMID, Julian H. **The Seebeck and Peltier Effects**, 2017. Disponível em:
< <http://iopscience.iop.org/book/978-1-6817-4641-8/chapter/bk978-1-6817-4641-8ch1> />. Acesso em 25 de março de 2018

HAMPSON, Geoff. **How a Thermistor Works**, 201-. Disponível em:

< <https://www.kitronik.co.uk/blog/how-a-thermistor-works/> >. Acesso em 25 de março de 2018

HYPER PHYSICS. **Hall Effect**, 201-. Disponível em:

< <http://hyperphysics.phy-astr.gsu.edu/hbase/magnetic/Hall.html> >. Acesso em: 16 de setembro 2018

KANDA, Denise; ROZÁRIO Mayara. **Mercado vegetariano: confira as principais mudanças do ramo no Brasil**, 2017. Disponível em:

< <http://economia.ig.com.br/2017-04-12/mercado-vegetariano.html> >. Acesso em: 20 de março 2018

KRON, Kron Instrumentos Elétricos Ltda. **Transdutor de Tensão ou Corrente Alternada (Rev. 3.0)**. - Manual Técnico, MARÇO 2010

LIOHM. **Sensor de Umidade e Temperatura - FHT20 Series**, 2019. Disponível em:

< <https://liohm.com.br/> >. Acesso em: 13 de março 2019

MILHORANCE, Flávia. **Cresce a procura por alimentos saudáveis no país, segundo relatório**, 2014. Disponível em:

< <https://oglobo.globo.com/sociedade/sustentabilidade/cresce-procura-por-alimentos-saudaveis-no-pais-segundo-relatorio-12721587> >. Acesso em: 20 de março 2018

MOTA, Thaís. **Termopar ou termorresistência?**, 2017. Disponível em:

< <http://blog.wika.com.br/know-how/termopar-ou-termorresistencia/> >. Acesso em 25 de março de 2018

MULTIFRIO. **Unidade Condensadora Hermética - Média - Danfoss**, 2019. Disponível em: < <http://www.multifrio.com.br/> >. Acesso em: 16 de março 2019

NIEUWENHUIJZEN, Bram van, **Small-scale mushroom cultivation - 2**, 1a edição, Agromisa Foundation and CTA, Wageningen, 2007

PIRES, Denise Prazeres Lopes; AFONSO, Júlio Carlos, CHAVES, Francisco Artur Braun, **A termometria nos séculos XIX e XX** - Revista Brasileira de Ensino de Física, Vol. 28 N° 1, JANEIRO 2006

PROSOLSOLAR. **Produtos**, 2019. Disponível em:

< <http://prosolsolar.com.br/produtos-reservatorios/> >. Acesso em: 19 de novembro 2019

RASPBERRY PI. **Raspberry Pi 3 Model B**, 2019. Disponível em:

< <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/> />. Acesso em: 23 de junho 2019

REVISTA SOCIEDADE DA MESA. **Cogumelo é excelente fonte de proteína**, 2012. Disponível em: < <http://revista.sociedadedamesa.com.br/2017/05/cogumelo-e-excelente-fonte-de-proteina/> />. Acesso em: 20 de março 2018

ROGERS, Bandom, **10 Questions to Ask Before Selecting a Humidity Sensor**, 2015. Disponível em: < <https://www.setra.com/blog/10-questions-to-ask-before-selecting-a-humidity-sensor/> />. Acesso em: 31 de março 2018

SEIDEL, Álysson Raniere. **Instrumentação Aplicada**. Santa Maria: Colégio Técnico Industrial de Santa Maria, 2011. p. 57. Material Didático – Sistema Escola Técnica Aberta do Brasil e-Tec Brasil, Santa Maria, 2011.

TEIXEIRA, Gustavo Grillo, (2017). In: PORTELA, Tatiana. **Com tecnologia própria, amigos cultivam cogumelos em aldeia**, 2017. Disponível em:

< <https://poraqui.news/aldeia/com-tecnologia-propria-amigos-cultivam-cogumelos-em-aldeia/> />. Acesso em: 20 de março 2018

TOMAZELA, José Maria. **Alta tecnologia em cogumelos**, 2017. Disponível em:

< <http://www.estadao.com.br/noticias/geral,alta-tecnologia-em-cogumelos,544031> />. Acesso em: 20 de março 2018

UNIVERSITY OF HAWAI'I. **Moving History**, 2019. Disponível em:

< <http://www.hawaii.edu/centennial/history.html> />. Acesso em: 20 de fevereiro 2019

VARGAS, Alexandre Maia. **Análise Diagnóstica da Cadeia Produtiva de Cogumelos do Distrito Federal**. Brasília: Universidade de Brasília, 2011. p. 84. Dissertação de Mestrado - Programa de Pós-Graduação em Agronegócios, Faculdade de Agronomia e Medicina Veterinária, Brasília, 2011.

APÊNDICE A - CÓDIGO DO SOFTWARE MAIN.PY

```
#!/usr/bin/env python
import smbus
import time
import RPi.GPIO as gpio
import minimalmodbus
import serial
import csv
from datetime import datetime

# bibliotecas para envio de email
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
import smtplib

#Configura a GPIO
gpio.setmode(gpio.BCM)
gpio.setwarnings(False)
gpio.setup(6, gpio.OUT)
gpio.setup(13, gpio.OUT)
gpio.setup(19, gpio.OUT)
gpio.setup(26, gpio.OUT)

TT01 = smbus.SMBus(3)
TT02 = smbus.SMBus(4)

#Ar Condicionado
gpio.output(26, gpio.HIGH)
#Irrigacao
gpio.output(19, gpio.HIGH)
#Ventilador
gpio.output(13, gpio.HIGH)
#Reserva
gpio.output(6, gpio.HIGH)

#portname, slave address (in decimal)
instrument = minimalmodbus.Instrument('/dev/ttyUSB0', 0)

def envia_email(arg0, arg1, arg2):
    # create message object instance
    msg = MIMEMultipart()
    message = arg0

    # setup the parameters of the message
    password = "*****"
    msg['From'] = "kogemelos@gmail.com"
    msg['To'] = arg1
    msg['Subject'] = arg2
    # add in the message body
    msg.attach(MIMEText(message, 'plain'))
    #create server
    server = smtplib.SMTP('smtp.gmail.com: 587')
    server.starttls()
    # Login Credentials for sending the mail
    server.login(msg['From'], password)
    # send the message via the server.
    server.sendmail(msg['From'], msg['To'], msg.as_string())
    server.quit()

def initial_config():
    now = datetime.now()
    dt_string = now.strftime("%d/%m/%Y %H:%M:%S")

    print("Programa Iniciado!")
```

```

print("Data/Hora",dt_string)
print("")

#print(instrument.serial.port
instrument.serial.baudrate = 9600
instrument.serial.bytesize = 8
instrument.serial.parity = serial.PARITY_NONE
instrument.serial.stopbits = 1
instrument.serial.timeout = 1

#print(instrument.address
instrument.mode = minimalmodbus.MODE_RTU # rtu or ascii mode
instrument.clear_buffers_before_each_transaction = True

def read_instruments():
    now = datetime.now()
    dt_string = now.strftime("%d/%m/%Y %H:%M:%S")

    TT01.write_i2c_block_data(0x5C,0x03,[0x00,0x02])
    block = TT01.read_i2c_block_data(0x5C,0x00,4)
    humidity_01 = (block[2] << 8 | block[3])/10.0

    TT01.write_i2c_block_data(0x5C,0x03,[0x02,0x02])
    block = TT01.read_i2c_block_data(0x5C,0x00,4)
    temperature_01 = (block[2] << 8 | block[3])/10.0

    TT02.write_i2c_block_data(0x5C,0x03,[0x00,0x02])
    block = TT02.read_i2c_block_data(0x5C,0x00,4)
    humidity_02 = (block[2] << 8 | block[3])/10.0

    TT02.write_i2c_block_data(0x5C,0x03,[0x02,0x02])
    block = TT02.read_i2c_block_data(0x5C,0x00,4)
    temperature_02 = (block[2] << 8 | block[3])/10.0

    ## Read CO2
    co2 = instrument.read_register(108, 0) # Registernumber, number of
decimals

    print("Data/Hora",dt_string)
    print("TT01 Temperature: " + str(temperature_01) + "°C")
    print("TT02 Temperature: " + str(temperature_02) + "°C")
    print("HT01 Humidity: " + str(humidity_01) + "%")
    print("HT02 Humidity: " + str(humidity_02) + "%")
    print("Nivel de CO2: " + str(co2) + "ppm" + "\n")

    #Confirmar se precisa gravar esse arquivo
    #with open('/home/pi/Codes/dados_estufa01.csv','a') as fd:
    #    fd.write(dt_string + "," + str(co2) + "," + str(temperature_01) +
", " + str(humidity_01) + "," + str(temperature_02) + "," + str(humidity_02)
+ "," + "0.0" + "," + "0" + "," + "0" + "," + "0" + "," + "0" + "," + "0" + "\n")

    with open('/home/pi/Servidor/public/db_medidas2.csv','a') as fd:
        fd.write(dt_string + "," + str(co2) + "," + str(temperature_01) + ","
+ str(humidity_01) + "," + str(temperature_02) + "," + str(humidity_02) +
", " + str(consumption) + "," + "0" + "," + "0" + "," + "0" + "," + "0" +
"\n")

def controller():
    #Controle do Ar Condicionado
    global enable_umid
    global enable_hvac

    TT01.write_i2c_block_data(0x5C,0x03,[0x00,0x02])
    block = TT01.read_i2c_block_data(0x5C,0x00,4)
    humidity_01 = (block[2] << 8 | block[3])/10.0

```

```

TT01.write_i2c_block_data(0x5C,0x03,[0x02,0x02])
block = TT01.read_i2c_block_data(0x5C,0x00,4)
temperature_01 = (block[2] << 8 | block[3])/10.0

TT02.write_i2c_block_data(0x5C,0x03,[0x00,0x02])
block = TT02.read_i2c_block_data(0x5C,0x00,4)
humidity_02 = (block[2] << 8 | block[3])/10.0

TT02.write_i2c_block_data(0x5C,0x03,[0x02,0x02])
block = TT02.read_i2c_block_data(0x5C,0x00,4)
temperature_02 = (block[2] << 8 | block[3])/10.0

## Read CO2
# Registernumber, number of decimals
co2 = instrument.read_register(108, 0)

#Read clock
now = datetime.now()
dt_string = now.strftime("%d/%m/%Y %H:%M:%S")

#Read Configs
with open('/home/pi/Codes/config.txt','r') as fd:
    vt_lines = fd.readlines()

#Save the consumption information
consumption = round(float(vt_lines[2]),1)

#Habilita funcionamento dos ventiladores
if co2 > 900:
    #Habilita Ventiladores
    gpio.output(13,gpio.LOW)

#Desabilita funcionamento dos ventiladores
if co2 < 700:
    #Desabilita Ventiladores
    gpio.output(13,gpio.HIGH)

if humidity_01 > float("98.0"):
    #Ativa o Ar Condicionado
    gpio.output(26,gpio.LOW)
    enable_umid = 24
    print("Ar condicionado ligado HT01++")
    with open('/home/pi/Codes/report.txt','a') as fd:
        fd.write(dt_string + " - Ar condicionado ativado devido a alta
umidade" + "\n")
    #Envia email
    try:
        message = "Ar condicionado ativado devido à alta umidade" + "\n" +
"Temp.Int= " + str(temperature_01) + "\n" + "Umid.Int= " + str(humidity_01)
+ "\n" + "Temp.Ext= " + str(temperature_02) + "\n" + "Umid.Ext= " +
str(humidity_02) + "\n" + "CO2= " + str(co2)
        envia_email(message,"kogeratski.ufpr@gmail.com","Estufa 01 - Inf")
        envia_email(message,"jessyteter@gmail.com","Estufa 01 - Inf")
    except Exception as e:
        pass
else:
    #Mantem o ar condicionado ativado por tempo mínimo
    if enable_umid > 0:
        with open('/home/pi/Codes/report.txt','a') as fd:
            fd.write(dt_string + " - Enable Umid -1" + "\n")
            enable_umid = enable_umid - 1

#Habilita o funcionamento do HVAC
if (temperature_02 < float(vt_lines[0])) and (gpio.input(26) == 1) and
(enable_hvac == 0) and (temperature_01 < float(vt_lines[0])):

```

```

enable_hvac = int(vt_lines[1])

#Liga o HVAC se estiver desligado e habilitado
if (gpio.input(26) == 1) and (enable_hvac > 0):
    #Ativa o Ar Condicionado
    gpio.output(26,gpio.LOW)
    print ("Ar condicionado ligado TT01 < SP")
    with open('/home/pi/Codes/report.txt','a') as fd:
        fd.write(dt_string + " - Ar condicionado ativado devido a baixa
temperatura" + "\n")
    #Envia email
    try:
        message = "Ar condicionado ativado devido a baixa temperatura" +
"\n" + "Temp.Int= " + str(temperature_01) + "\n" + "Umid.Int= " +
str(humidity_01) + "\n" + "Temp.Ext= " + str(temperature_02) + "\n" +
"Umid.Ext= " + str(humidity_02) + "\n" + "CO2= " + str(co2)
        envia_email(message,"kogeratski.ufpr@gmail.com","Estufa 01 - Inf")
        envia_email(message,"jessyteter@gmail.com","Estufa 01 - Inf")
    except Exception as e:
        pass

#Garante funcionamento do HVAC por tempo mínimo
if (temperature_01 > float(vt_lines[0])) and (gpio.input(26) == 0) and
(temperature_02 > float(vt_lines[0]) - 4.0):
    if enable_hvac > 0:
        with open('/home/pi/Codes/report.txt','a') as fd:
            fd.write(dt_string + " - Enable HVAC -1" + "\n")
            enable_hvac = enable_hvac - 1

#Desliga o HVAC se estiver ligado e TT02 > SP e TT01 > SP
if (temperature_02 > float(vt_lines[0]) + 0.5) and (gpio.input(26) == 0)
and (temperature_01 > float(vt_lines[0])):
    #Desativa o Ar Condicionado
    gpio.output(26,gpio.HIGH)
    enable_hvac = 0
    print ("Ar condicionado ligado TT02 > SP")
    with open('/home/pi/Codes/report.txt','a') as fd:
        fd.write(dt_string + " - Ar condicionado desativado devido a
temperatura externa > SP" + "\n")

    #Envia email
    try:
        message = "Ar condicionado desativado devido a temperatura externa
> SP" + "\n" + "Temp.Int= " + str(temperature_01) + "\n" + "Umid.Int= " +
str(humidity_01) + "\n" + "Temp.Ext= " + str(temperature_02) + "\n" +
"Umid.Ext= " + str(humidity_02) + "\n" + "CO2= " + str(co2)
        envia_email("Ar condicionado desativado devido a temperatura
externa > SP","kogeratski.ufpr@gmail.com","Estufa 01 - Inf")
        envia_email("Ar condicionado desativado devido a temperatura
externa > SP","jessyteter@gmail.com","Estufa 01 - Inf")
    except Exception as e:
        pass

#Desliga o HVAC
if (enable_hvac == 0 and enable_umid == 0 and gpio.input(26) == 0):
    #Desativa o Ar Condicionado
    gpio.output(26,gpio.HIGH)
    print ("Ar condicionado desligado TT01 e HT01 ok")
    with open('/home/pi/Codes/report.txt','a') as fd:
        fd.write(dt_string + " - Ar condicionado desativado" + "\n")

    #Envia email
    try:
        message = "Ar condicionado desativado" + "\n" + "Temp.Int= " +
str(temperature_01) + "\n" + "Umid.Int= " + str(humidity_01) + "\n" +

```

```

Temp.Ext= " + str(temperature_02) + "\n" + "Umid.Ext= " + str(humidity_02)
+ "\n" + "CO2= " + str(co2)
    envia_email(message, "kogeratski.ufpr@gmail.com", "Estufa 01 - Inf")
    envia_email(message, "jessyteter@gmail.com", "Estufa 01 - Inf")
    except Exception as e:
        pass

initial_config()
enable_hvac = 0
enable_umid = 0
consumption = 0.0

time.sleep(1)
#Envia email
try:
    message = "Sistema Iniciado!"
    envia_email(message, "kogeratski.ufpr@gmail.com", "Estufa 01 - Inf")
    envia_email(message, "jessyteter@gmail.com", "Estufa 01 - Inf")
except Exception as e:
    pass

while (True):
    time.sleep(1)

    try:
        #Faz a leitura dos instrumentos
        read_instruments()
        #Faz o acionamento dos atuadores
        controller()
    except Exception as e:
        pass

    time.sleep(299)

```

APÊNDICE B - CÓDIGO DO SERVIDOR NODE_SERVER.JS

```
var express = require('express');
var app = express();
var path = require('path');
app.use(express.static('public'));

app.get('/write', function (req, res) {
  res.send('The file has been written!');
  var fs = require('fs');
  var stream = fs.createWriteStream("my_file.txt");
  stream.once('open', function (fd) {
    stream.write("My first row\n");
    stream.write("My second row\n");
    stream.end();
  });
});

app.get('/update_ios', function (req, res) {
  var net = require('net');

  var client = new net.Socket();
  client.connect(9999, '192.168.1.2', function() {
    console.log('Connected');
    client.write('Hello, server! Love, Client. ');
    client.destroy(); // kill client after server's response
  });

  client.on('close', function() {
    console.log('Connection closed');
  });
});

app.get('/test/', function (req, res) {
  res.send('id: ' + req.query.id);
});

app.get("/insert", function (req, res) {
  const fs = require('fs');
  res.send('The data has been updated!');
});

app.get('/', function (req, res) {
  //res.sendFile(__dirname + '/index.html');
  res.sendFile(__dirname + '/Main.html');
});

var server = app.listen(8081, function () {
  var host = server.address().address
  var port = server.address().port

  console.log("Example app listening at http://%s:%s", host, port)
});
```

APÊNDICE C - CÓDIGO DA PÁGINA HTML INDEX.HTML

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=0.7,
maximum-scale=1.0, user-scalable=no" />
</head>

<style>

.vertical_space {
  margin-top: 0.5cm;
}

.curva1 {
  fill: none;
  stroke: Red;
  stroke-linejoin: round;
  stroke-linecap: round;
  stroke-width: 1;
  clip-path: url(#clip);
}

.curva2 {
  fill: none;
  stroke: Blue;
  stroke-linejoin: round;
  stroke-linecap: round;
  stroke-width: 1;
  clip-path: url(#clip);
}

.curva3 {
  fill: none;
  stroke: DarkOliveGreen;
  stroke-linejoin: round;
  stroke-linecap: round;
  stroke-width: 0.3;
  clip-path: url(#clip);
}

.curva4 {
  fill: none;
  stroke: Red;
  stroke-linejoin: round;
  stroke-linecap: round;
  stroke-dasharray: 2,5;
  stroke-width: 1;
  clip-path: url(#clip);
}

.curva5 {
  fill: none;
  stroke: Blue;
  stroke-linejoin: round;
  stroke-linecap: round;
  stroke-dasharray: 2,5;
  stroke-width: 1;
  clip-path: url(#clip);
}
```

```

.curva6 {
    fill: none;
    stroke: DarkGreen;
    stroke-linejoin: round;
    stroke-linecap: round;
    stroke-width: 1;
    clip-path: url(#clip);
}

.zoom {
    cursor: move;
    fill: none;
    pointer-events: all;
}

.axisRed line{
    stroke: Red;
}

.axisRed path{
    stroke: Red;
}

.axisRed text{
    fill: Red;
}

.axisBlue line{
    stroke: Blue;
}

.axisBlue path{
    stroke: Blue;
}

.axisBlue text{
    fill: Blue;
}

.axisGreen line{
    stroke: DarkGreen;
}

.axisGreen path{
    stroke: DarkGreen;
}

.axisGreen text{
    fill: DarkGreen;
}

/* Customize the label (the container) */
.container {
    display: block;
    position: relative;
    padding-left: 0px;
    margin-bottom: 12px;
    cursor: pointer;
    font-size: 16px;
    -webkit-user-select: none;
    -moz-user-select: none;
    -ms-user-select: none;
    user-select: none;
}

/* Estilo dos botões de comando */

```

```

/* Hide the browser's default radio button */
.container input {
  position: absolute;
  opacity: 0;
  cursor: pointer;
}

/* Create a custom radio button */
.checkmark {
  position: absolute;
  top: 0;
  left: 0;
  height: 25px;
  width: 25px;
  background-color: #eee;
  border-radius: 50%;
}

/* On mouse-over, add a grey background color */
.container:hover input ~ .checkmark {
  background-color: #ccc;
}

/* When the radio button is checked, add a blue background */
.container input:checked ~ .checkmark {
  background-color: #2196F3;
}

/* Create the indicator (the dot/circle - hidden when not checked) */
.checkmark:after {
  content: "";
  position: absolute;
  display: none;
}

/* Show the indicator (dot/circle) when checked */
.container input:checked ~ .checkmark:after {
  display: block;
}

/* Style the indicator (dot/circle) */
.container .checkmark:after {
  top: 9px;
  left: 9px;
  width: 8px;
  height: 8px;
  border-radius: 50%;
  background: white;
}

/* Estilo das colunas*/
.column1 {
  float: left;
  width: 130px;
  padding: 4px;
}

.column2 {
  float: left;
  width: 100px;
  padding: 4px;
}

.column3 {
  float: left;
}

```

```

    width: 100px;
    padding: 4px;
}

.column4 {
    float: left;
    width: 130px;
    padding: 4px;
}

.column5 {
    float: left;
    width: 100px;
    padding: 4px;
}

/* Clear floats after the columns */
.row:after {
    content: "";
    display: table;
    clear: both;
}

/* Estilo dos inputs*/
input[type=text], select {
    width: 80%;
    padding: 8px 20px;
    display: inline-block;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}

input[type=submit] {
    width: 80%;
    background-color: #4CAF50;
    color: white;
    padding: 14px 0px;
    margin: 8px 0px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

input[type=submit]:hover {
    background-color: #45a049;
}
</style>

<body style="width: 1300px">

<!-- Detecta a largura da Janela -->
<body>

    <!-- Limita a largura da página -->
    <p><b><u> Sistema Supervis&#243;rio para Monitoramento de Estufas de
Cogumelos </u></b></p>
    <p> Gr&#225;fico 01 - Temperatura e Umidade Relativa do Ar</p>

    <svg id= "graf_01" width="1300" height="250"></svg>
    <div class="row">
        <div class="col-sm-2">
            <button type="button" onclick= zoom_in()>Zoom (+)</button>
            <button type="button" onclick= zoom_out()>Zoom (-)</button>
        </div>

```

```

</div>

<p> Gr&#225;fico 02 - Nivel de CO2</p>
<svg id= "graf_02" width="1280" height="250"></svg>
<div class="row">
  <div class="col-sm-2">
    <button type="button" onclick= zoom_in()>Zoom (+)</button>
    <button type="button" onclick= zoom_out()>Zoom (-)</button>
  </div>
</div>

<script
src="https://cdnjs.cloudflare.com/ajax/libs/d3/4.13.0/d3.min.js"></script>
<script src="https://hammerjs.github.io/dist/hammer.js"></script>

<script>
var inter = setInterval(function() {
  updateData();
  //ao0_update();
}, 5000);

  //SVG Grafico 01
var svg1 = d3.select("#graf_01"),
  margin = {top: 20, right: 20, bottom: 30, left: 40},
  margin2 = {top: 430, right: 20, bottom: 30, left: 40},
width = +svg1.attr("width") - margin.left - margin.right -
10,
  height = +svg1.attr("height") - margin.top - margin.bottom;

  //SVG Grafico 02
var svg2 = d3.select("#graf_02"),
  margin = {top: 20, right: 20, bottom: 30, left: 40},
  margin2 = {top: 430, right: 20, bottom: 30, left: 40},
width = +svg2.attr("width") - margin.left - margin.right -
10,
  height = +svg2.attr("height") - margin.top - margin.bottom;

var parseDate = d3.timeParse("%e/%m/%Y %H:%M:%S");

// Cria o eixo x dos gráficos
var x = d3.scaleTime().range([0, width]),
  x2 = d3.scaleTime().range([0, width]),

// Cria o eixo vertical da temperatura
  y1 = d3.scaleLinear().range([height, 0]);
// Cria o eixo vertical da umidade
  y2 = d3.scaleLinear().range([height, 0]);
// Cria o eixo vertical do CO2
  y3 = d3.scaleLinear().range([height, 0]);

//Ajusta a posição os eixos
  var xAxis = d3.axisBottom(x);
  var y1Axis = d3.axisLeft(y1);
  var y2Axis = d3.axisRight(y2);
  var y3Axis = d3.axisLeft(y3).scale(y3).ticks(5);

//Define as variáveis do Zoom
  var zoom1 = d3.zoom() //Grafico 01
    .scaleExtent([1, Infinity])
    .translateExtent([[0, 0], [width, height]])
    .extent([[0, 0], [width, height]])
    .on("zoom", zoomed1);

var zoom2 = d3.zoom() //Grafico 02
  .scaleExtent([1, Infinity])
  .translateExtent([[0, 0], [width, height]])

```

```

        .extent([[0, 0], [width, height]])
        .on("zoom", zoomed2);

//Hammer
var hammer_obj = document.getElementById('graf_01');
var mc = new Hammer.Manager(hammer_obj);
var pinch = new Hammer.Pinch();
mc.on("pinch", zoom_in());
//End Hammer

var curva1 = d3.area()//TempInt
    .x(function(d) { return x(d.Date); })
    .y0(function(d) { return y1(d.TempInt); })
    .y1(function(d) { return y1(d.TempInt); });

var curva2 = d3.area()//UmidInt
    .x(function(d) { return x(d.Date); })
    .y0(function(d) { return y2(d.UmidInt); })
    .y1(function(d) { return y2(d.UmidInt); });

//
var curva3 = d3.area()
//
    .x(function(d) { return x(d.Date); })
//
    .y0(function(d) { return y(d.tpo); })
//
    .y1(function(d) { return y(d.tpo); });

var curva4 = d3.area()//TempExt
    .x(function(d) { return x(d.Date); })
    .y0(function(d) { return y1(d.TempExt); })
    .y1(function(d) { return y1(d.TempExt); });

var curva5 = d3.area()//UmidExt
    .x(function(d) { return x(d.Date); })
    .y0(function(d) { return y2(d.UmidExt); })
    .y1(function(d) { return y2(d.UmidExt); });

var curva6 = d3.area()//CO2
    .x(function(d) { return x(d.Date); })
    .y0(function(d) { return y3(d.CO2); })
    .y1(function(d) { return y3(d.CO2); });

//Cria um limitador que confina o gráfico dentro dele
svg1.append("defs").append("clipPath")
    .attr("id", "clip")
    .append("rect")
    .attr("width", width)
    .attr("height", height);

svg2.append("defs").append("clipPath")
    .attr("id", "clip")
    .append("rect")
    .attr("width", width)
    .attr("height", height);

//Define os elementos que serão anexados no gráfico 01
var focusA1 = svg1.append("g")
    .attr("class", "focus")
    .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

var focusA2 = svg1.append("g")
    .attr("class", "focus")
    .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

var focusA3 = svg1.append("g")
    .attr("class", "focus")

```

```

        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

    var focusA4 = svg1.append("g")
        .attr("class", "focus")
        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

    var focusA5 = svg1.append("g")
        .attr("class", "focus")
        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

    var backline1 = svg1.append("line")
        .style("stroke", "Black")
        .style("stroke-width", 0.1)
        .attr("x1", 41)
        .attr("y1", 20.5)
        .attr("x2", 1250)
        .attr("y2", 20.5);

    var backline2 = svg1.append("line")
        .style("stroke", "Black")
        .style("stroke-width", 0.1)
        .attr("x1", 41)
        .attr("y1", 60.5)
        .attr("x2", 1250)
        .attr("y2", 60.5);

    var backline3 = svg1.append("line")
        .style("stroke", "Black")
        .style("stroke-width", 0.1)
        .attr("x1", 41)
        .attr("y1", 100.5)
        .attr("x2", 1250)
        .attr("y2", 100.5);

    var backline4 = svg1.append("line")
        .style("stroke", "Black")
        .style("stroke-width", 0.1)
        .attr("x1", 41)
        .attr("y1", 140.5)
        .attr("x2", 1250)
        .attr("y2", 140.5);

    var backline5 = svg1.append("line")
        .style("stroke", "Black")
        .style("stroke-width", 0.1)
        .attr("x1", 41)
        .attr("y1", 180.5)
        .attr("x2", 1250)
        .attr("y2", 180.5);

    //    var context = svg1.append("g")
    //        .attr("class", "context")
    //        .attr("transform", "translate(" + margin2.left + "," +
margin2.top + ")");

    //Define os elementos que serão anexados no gráfico 02
    var focusB1 = svg2.append("g")
        .attr("class", "focus")
        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")");

    var focusB2 = svg2.append("g")
        .attr("class", "focus")

```

```

        .attr("transform", "translate(60," + margin.top + ")");

var backline6 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 20.5)
  .attr("x2", 1250)
  .attr("y2", 20.5);

var backline7 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 53.8)
  .attr("x2", 1250)
  .attr("y2", 53.8);

var backline8 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 87.1)
  .attr("x2", 1250)
  .attr("y2", 87.1);

var backline9 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 120.5)
  .attr("x2", 1250)
  .attr("y2", 120.5);

var backline10 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 153.9)
  .attr("x2", 1250)
  .attr("y2", 153.9);

var backline11 = svg2.append("line")
  .style("stroke", "Black")
  .style("stroke-width", 0.1)
  .attr("x1", 41)
  .attr("y1", 187.3)
  .attr("x2", 1250)
  .attr("y2", 187.3);

//      var context = svg2.append("g")
//          .attr("class", "context")
//          .attr("transform", "translate(" + margin2.left + "," +
margin2.top + ")");

d3.csv("http://192.168.1.2:8081/db_medidas2.csv", type, function(error,
data) {
    //if (error) throw error;

    x.domain(d3.extent(data, function(d) { return d.Date;}));
    y1.domain([10, 35]);
    x2.domain(x.domain());
    y2.domain([0, 100]);
    y3.domain([0, 1.2]);

```

```

//Anexa os elementos ao gráfico 01
focusA1.append("path")
    .datum(data)
    .attr("class", "curval1")
    .attr("d", curval1);

focusA2.append("path")
    .datum(data)
    .attr("class", "curva2")
    .attr("d", curva2);

//
// focusA3.append("path")
//     .datum(data)
//     .attr("class", "curva3")
//     .attr("d", curva3);

focusA4.append("path")
    .datum(data)
    .attr("class", "curva4")
    .attr("d", curva4);

focusA5.append("path")
    .datum(data)
    .attr("class", "curva5")
    .attr("d", curva5);

focusA1.append("g")
    .attr("class", "axis axis--x")
    .attr("transform", "translate(0," + height + ")")
    .call(xAxis);

focusA1.append("text")
    .attr("transform", "rotate(-90)")
    .attr("y", 0 - margin.left)
    .attr("x", 0 - 40)
    .attr("dy", "1em")
    .style("text-anchor", "middle")
    .attr("font-size", "12px")
    .style('font-family', 'CrimsonItalic')
    .text("Temperatura [\u00B0C]");

focusA1.append("text")
    .attr("transform", "rotate(-90)")
    .attr("y", 1240)
    .attr("x", 0 - 30)
    .attr("dy", "1em")
    .style("text-anchor", "middle")
    .attr("font-size", "12px")
    .style('font-family', 'CrimsonItalic')
    .text("Umidade [%]");

focusA1.append("g")
    .attr("class", "axisRed")
    .attr("transform", "translate(0,0)")
    .call(y1Axis);

focusA1.append("g")
    .attr("class", "axisBlue")
    .attr("transform", "translate(1210,0)")
    .call(y2Axis);

svg1.append("rect")
    .attr("class", "zoom")
    .attr("width", width)
    .attr("height", height)

```

```

        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")")
        .call(zoom1);

//Anexa os elementos ao gráfico 02
focusB1.append("path")
        .datum(data)
        .attr("class", "curva6")
        .attr("d", curva6);

//
focusB2.append("path")
//
        .datum(data)
//
        .attr("class", "curva6")
//
        .attr("d", curva6);

focusB1.append("g")
        .attr("class", "axis axis--x")
        .attr("transform", "translate(0," + height + ")")
        .call(xAxis);

focusB1.append("text")
        .attr("transform", "rotate(-90)")
        .attr("y", 0 - margin.left - 2)
        .attr("x", 0 - 62)
        .attr("dy", "1em")
        .style("text-anchor", "middle")
        .attr("font-size", "12px")
        .style('font-family', 'CrimsonItalic')
        .text("Nivel de CO2 [ppm x1000]");

//
focusB1.append("text")
//
        .attr("transform", "rotate(-90)")
//
        .attr("y", 1240)
//
        .attr("x", 0 - 30)
//
        .attr("dy", "1em")
//
        .style("text-anchor", "middle")
//
        .attr("font-size", "12px")
//
        .style('font-family', 'CrimsonItalic')
//
        .text("Umidade [%]");

focusB1.append("g")
        .attr("class", "axisGreen")
        .attr("transform", "translate(0,0)")
        .call(y3Axis);

//
focusB1.append("g")
//
        .attr("class", "axisBlue")
//
        .attr("transform", "translate(1210,0)")
//
        .call(y2Axis);

svg2.append("rect")
        .attr("class", "zoom")
        .attr("width", width)
        .attr("height", height)
        .attr("transform", "translate(" + margin.left + "," +
margin.top + ")")
        .call(zoom2);
});

function Window_width() { //Flag
    var w = window.outerWidth;
    var h = window.outerHeight;
    console.log(w);
}

function zoomed1() {

```

```

        if (d3.event.sourceEvent && d3.event.sourceEvent.type === "brush")
return; // ignore zoom-by-brush

    var t = d3.event.transform;
    x.domain(t.rescaleX(x2).domain());
    //console.log(t.rescaleX(x2).domain());
    focusA1.select(".curva1").attr("d", curva1); //Graf01
    focusA1.select(".axis--x").call(xAxis); //Eixo
    focusA2.select(".curva2").attr("d", curva2); //Graf02
    //focusA3.select(".curva3").attr("d", curva3); //Graf03
    focusA4.select(".curva4").attr("d", curva4); //Graf04
    focusA5.select(".curva5").attr("d", curva5); //Graf05
}

function zoomed2 () {
    if (d3.event.sourceEvent && d3.event.sourceEvent.type === "brush")
return; // ignore zoom-by-brush

    var t = d3.event.transform;
    x.domain(t.rescaleX(x2).domain());
    //console.log(t.rescaleX(x2).domain());
    focusB1.select(".curva6").attr("d", curva6); //Graf01
    focusB1.select(".axis--x").call(xAxis); //Eixo
}

function zoom_in () { //FLAG
    zoom1.scaleBy(svg1.transition().duration(2000), 2);
}

function zoom_out () { //FLAG
    zoom1.scaleBy(svg1.transition().duration(2000), 0.5);
}

function type(d) {
    d.Date = parseDate(d.Date);
    d.TempInt = +d.TempInt;
    d.UmidInt = +d.UmidInt;
    d.TempExt = +d.TempExt;
    d.UmidExt = +d.UmidExt;
    d.CO2 = +d.CO2 /1000;
    return d;
}

//Update graph
function updateData () {
    d3.csv("http://192.168.1.2:8081/db_medidas2.csv", type,
function(error, data) {
    if (error) throw error;

    x.domain(d3.extent(data, function(d) { return d.Date; }));
    x2.domain(x.domain());

    focusA1.select("path")
        .datum(data)

    focusA2.select("path")
        .datum(data)

//    focusA3.select("path")
//        .datum(data)

    focusA4.select("path")
        .datum(data)

    focusA5.select("path")

```

```

        .datum(data)

        focusB1.select("path")
            .datum(data)

        focusB2.select("path")
            .datum(data)

        svg1.select("rect")
            .attr("class", "zoom")
            .call(zoom1);
    });
}

</script>

<p></p>
<p></p>
<p></p>

<!--Line01-->
<div class="row">
  <div class="column1">
    <label class="container">Entrada Digital 01:</label>
  </div>

  <div class="column2">
    <label class="container">Desativada</label>
  </div>

  <div class="column3">
  </div>

  <div class="column4">
    <label class="container">Entrada Digital 05:</label>
  </div>

  <div class="column5">
    <label class="container">Desativada</label>
  </div>
</div>

<!--Line02-->
<div class="row">
  <div class="column1">
    <label class="container">Entrada Digital 02:</label>
  </div>

  <div class="column2">
    <label class="container">Desativada</label>
  </div>

  <div class="column3">
  </div>

  <div class="column4">
    <label class="container">Entrada Digital 06:</label>
  </div>

  <div class="column5">
    <label class="container">Desativada</label>
  </div>
</div>

```

```

<!Line03>
<div class="row">
  <div class="column1">
    <label class="container">Entrada Digital 03:</label>
  </div>

  <div class="column2">
    <label class="container">Desativada</label>
  </div>

  <div class="column3">
  </div>

  <div class="column4">
    <label class="container">Entrada Digital 07:</label>
  </div>

  <div class="column5">
    <label class="container">Desativada</label>
  </div>
</div>

<!Line04>
<div class="row">
  <div class="column1">
    <label class="container">Entrada Digital 04:</label>
  </div>

  <div class="column2">
    <label class="container">Desativada</label>
  </div>

  <div class="column3">
  </div>

  <div class="column4">
    <label class="container">Entrada Digital 08:</label>
  </div>

  <div class="column5">
    <label class="container">Desativada</label>
  </div>
</div>

<p class="vertical_space"></p>
<!Line01>
<div class="row">
  <div class="column2">
    <label class="container">Saída 01:</label>
  </div>

  <div class="column2">
    <label class="container">Ativar
      <input type="radio" checked="checked" name="radio1"
onclick="javascript:do1_high()">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column2">
    <label class="container">Desativar
      <input type="radio" checked="checked" name="radio1"
onclick="javascript:do1_low()">
      <span class="checkmark"></span>
    </label>
  </div>
</div>

```

```

</div>

<div class="column2">
  <label class="container">Saída 03:</label>
</div>

<div class="column2">
  <label class="container">Ativar
    <input type="radio" checked="checked" name="radio3"
onclick="javascript:do3_high()">
    <span class="checkmark"></span>
  </label>
</div>

<div class="column2">
  <label class="container">Desativar
    <input type="radio" checked="checked" name="radio3"
onclick="javascript:do3_low()">
    <span class="checkmark"></span>
  </label>
</div>

</div>

<!Line01>
<div class="row">
  <div class="column2">
    <label class="container">Saída 02:</label>
  </div>

  <div class="column2">
    <label class="container">Ativar
      <input type="radio" checked="checked" name="radio2"
onclick="javascript:do2_high()">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column2">
    <label class="container">Desativar
      <input type="radio" checked="checked" name="radio2"
onclick="javascript:do2_low()">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column2">
    <label class="container">Saída 04:</label>
  </div>

  <div class="column2">
    <label class="container">Ativar
      <input type="radio" checked="checked" name="radio4"
onclick="javascript:do4_high()">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column2">
    <label class="container">Desativar
      <input type="radio" checked="checked" name="radio4"
onclick="javascript:do4_low()">
      <span class="checkmark"></span>
    </label>
  </div>

```

```

</div>

<p class="vertical_space"></p>
<!Line01>
<div class="row">
  <div class="column3">
    <label class="container">Saída Analg. 01:</label>
  </div>

  <div class="column3">
    <label class="container">Automático
      <input type="radio" checked="checked" name="radio5">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <label class="container">Set Point
      <input type="radio" checked="checked" name="radio5">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <input type="text" id="box_a0" placeholder="000.0 ~100.0">
  </div>
</div>

<!Line01>
<div class="row">
  <div class="column3">
    <label class="container">Saída Analg. 02:</label>
  </div>

  <div class="column3">
    <label class="container">Automático
      <input type="radio" checked="checked" name="radio6">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <label class="container">Set Point
      <input type="radio" checked="checked" name="radio6">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <input type="text" id="box_a01" placeholder="000.0 ~100.0">
  </div>
</div>

<!Line01>
<div class="row">
  <div class="column3">
    <label class="container">Saída Analg. 03:</label>
  </div>

  <div class="column3">
    <label class="container">Automático
      <input type="radio" checked="checked" name="radio7">
      <span class="checkmark"></span>
    </label>
  </div>

```

```

<div class="column3">
  <label class="container">Set Point
    <input type="radio" checked="checked" name="radio7">
    <span class="checkmark"></span>
  </label>
</div>

<div class="column3">
  <input type="text" id="box_ao2" placeholder="000.0 ~100.0">
</div>
</div>

<!Line01>
<div class="row">
  <div class="column3">
    <label class="container">Saída Analg. 04:</label>
  </div>

  <div class="column3">
    <label class="container">Automático
      <input type="radio" checked="checked" name="radio8">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <label class="container">Set Point
      <input type="radio" checked="checked" name="radio8">
      <span class="checkmark"></span>
    </label>
  </div>

  <div class="column3">
    <input type="text" id="box_ao3" placeholder="000.0 ~100.0">
  </div>
</div>

<script>
  var xmlhttp = new XMLHttpRequest();
  var resp = ""

  xmlhttp.onreadystatechange = function() {

    if (xmlhttp.readyState==4 && xmlhttp.status==200) {
      resp = xmlhttp.responseText;
    }

    function do1_high() {
      xmlhttp.open("GET",
"http://192.168.1.2:8081/do01_high?id_estufa=Estufa_A", true);
      xmlhttp.send();

    }

    function do1_low() {
      xmlhttp.open("GET",
"http://192.168.1.2:8081/do01_low?id_estufa=Estufa_A", true);
      xmlhttp.send();
      //console.log(resp);
    }

    function do2_high() {
      xmlhttp.open("GET",
"http://192.168.1.2:8081/do02_high?id_estufa=Estufa_A", true);

```

```
        xmlhttp.send();
        //console.log(resp);
    }

    function do2_low() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do02_low?id_estufa=Estufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }

    function do3_high() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do03_high?id_estufa=Estufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }

    function do3_low() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do03_low?id_estufa=Estufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }

    function do4_high() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do04_high?id_estufa=Estufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }

    function do4_low() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do04_low?id_estufa=Estufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }

    function ao0_update() {
        xmlhttp.open("GET",
"http://192.168.1.2:8081/do04_low?ao0="+box_ao0.value+"Estufa_A&id_estufa=E
stufa_A", true);
        xmlhttp.send();
        //console.log(resp);
    }
}
</script>

</body>

</HTML>
```

APÊNDICE D - ALGORITMO E *MEMBERSHIP FUNCTIONS*

Para etapa futura o projeto prevê um controlador *Fuzzy* para manter a temperatura, umidade e nível de CO₂, que são as três variáveis mais significativas para o cultivo deste fungo. A escolha de um controlador *Fuzzy* se deu pela forma de parametrização do mesmo, que admite um certo grau de incerteza e pode ser configurado usando regras.

O escopo do projeto é limitar o produtor a definir os *set-points*, tolerâncias e definições das grandezas das variáveis (muito baixo, baixo, normal, alto e muito alto), mas para escopo futuro, os produtores poderiam até mesmo configurar as regras de controle. Cada produtor tem condições de fazer estes ajustes sem grandes dificuldades, porque isso faz parte da vivência deles. Caso fosse implementado um controlador PID, haveria uma dificuldade grande em fazer ajustes e parametrizações, pois as variáveis de controle não são de conhecimento dos produtores, o que seria uma barreira.

O algoritmo geral de controle é mostrado no 7.7 “Algoritmo de controle e *membership functions*”.

O controlador *Fuzzy* precisa das funções chamadas de “*Membership Functions*” ou MFs, que definem os estados das variáveis e seu grau de incerteza. As MFs definidas para o controlador em questão são mostradas nas figuras 38, 39, 40, 41 e 42 a seguir:

Figura 38 – MF da Temperatura e Umidade Interna

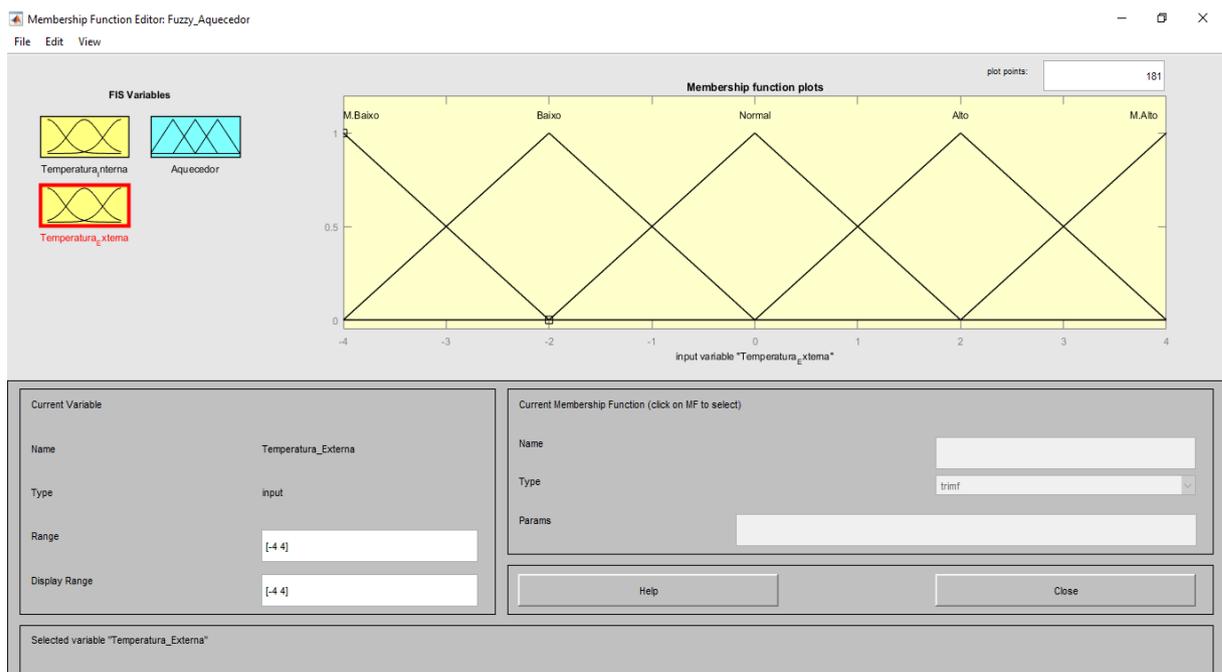
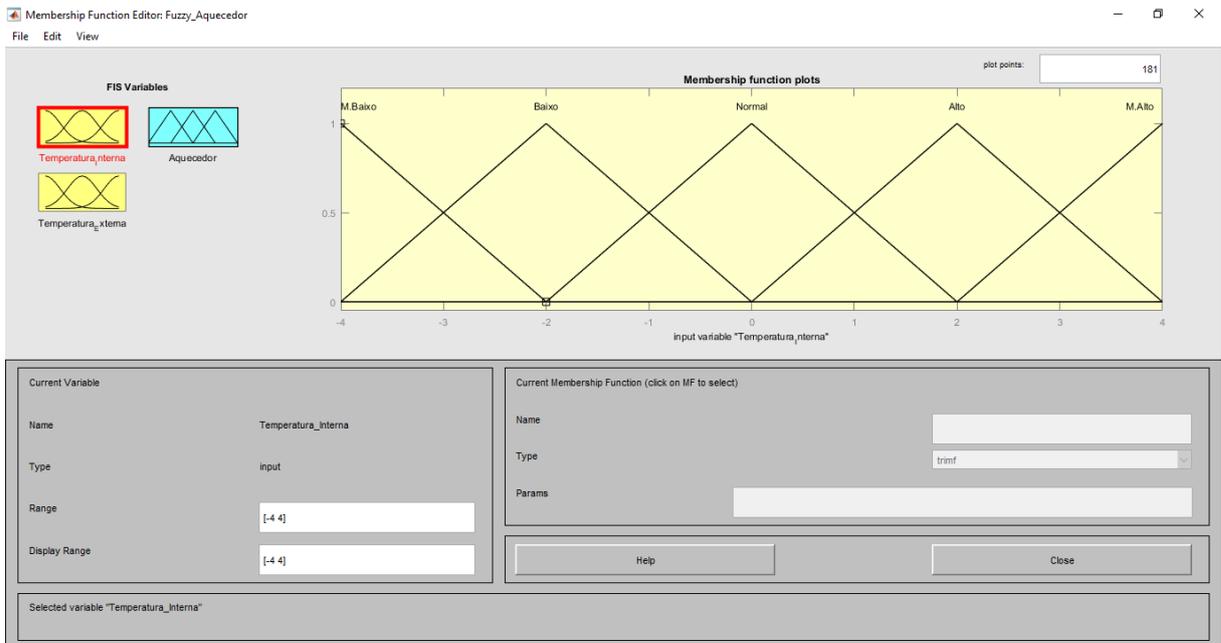
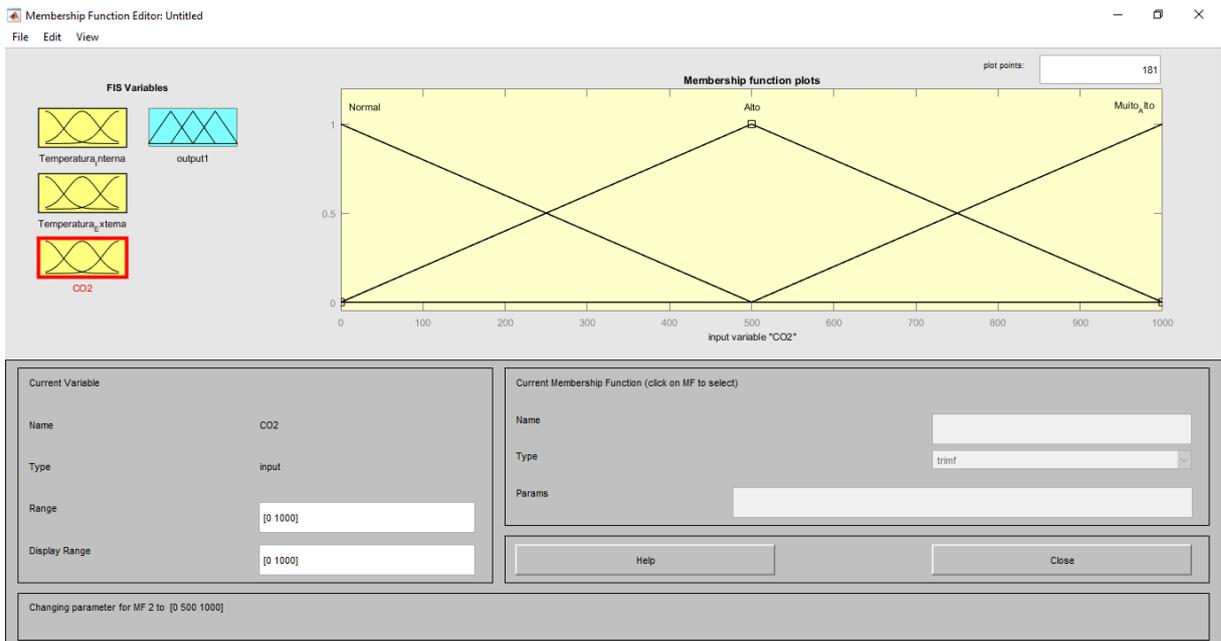


Figura 39 – MF da Temperatura e Umidade Externa



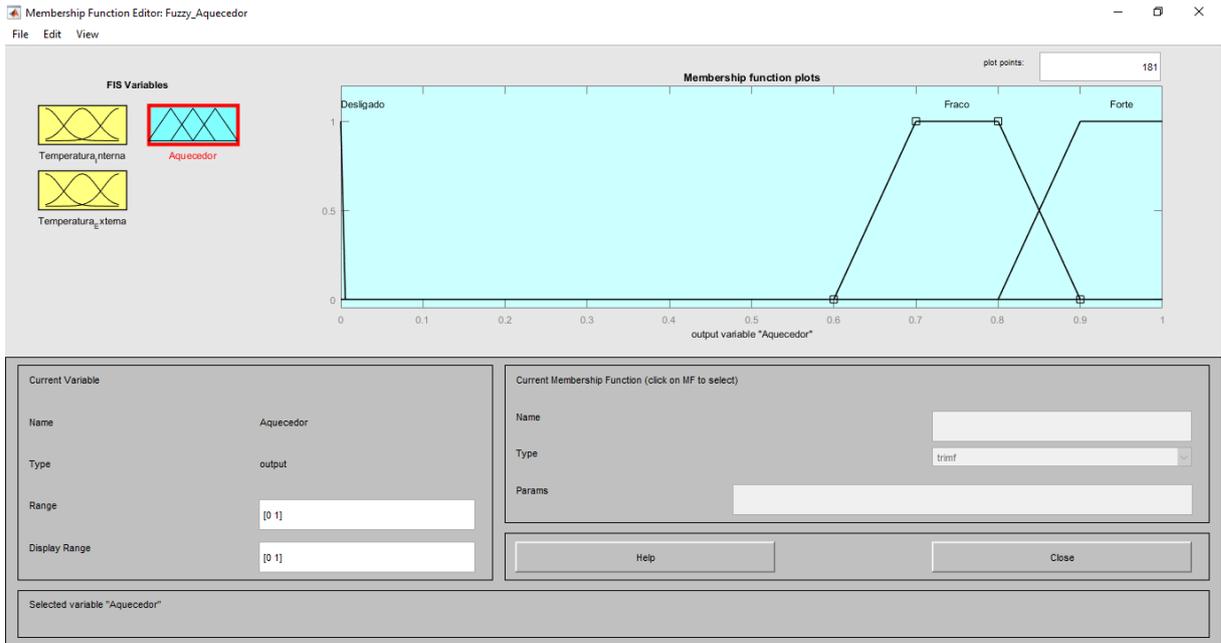
Fonte: 2019, Autor

Figura 40 – MF do Nível de CO₂



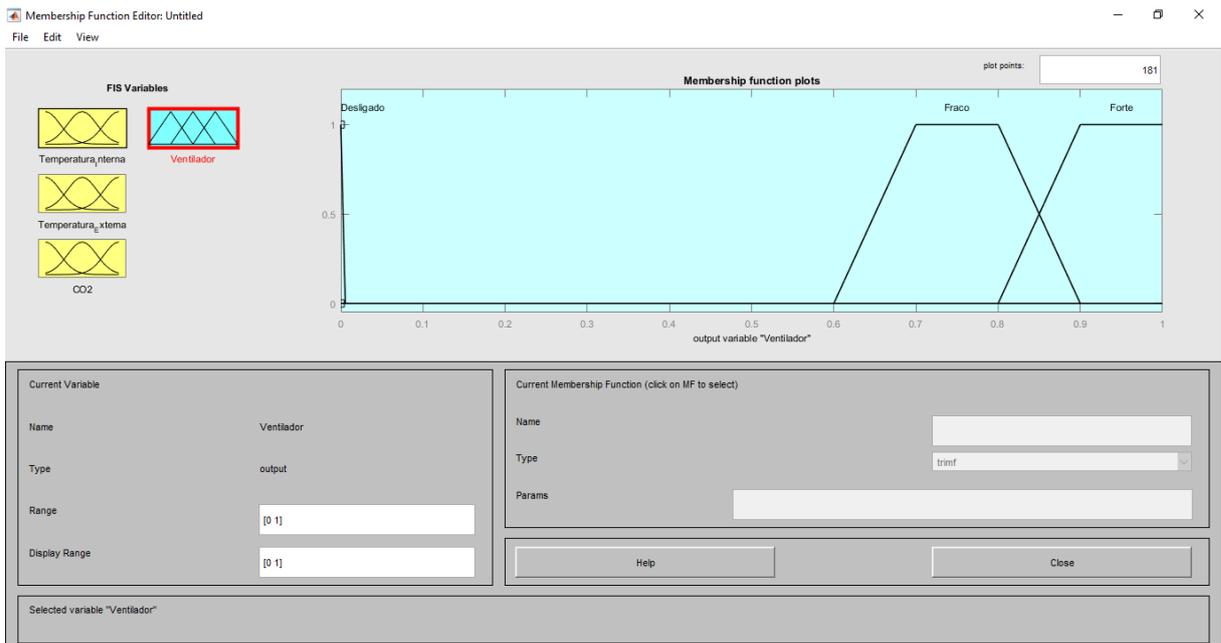
Fonte: 2019, Autor

Figura 41 – MF do Aquecedor e Resfriador



Fonte: 2019, Autor

Figura 42 – MF do Sistema de Ventilação



Fonte: 2019, Autor

As regras de controle foram feitas a partir do algoritmo de controle mostrado no 7.7 “Algoritmo de controle e *membership function*” e estão descritas a seguir:

Regras de Controle do Sistema de Aquecimento

Se Temperatura Interna = Normal o Aquecedor = Desligado;

Se Temperatura Interna = Baixa e Temperatura Externa = Muito Alta o Aquecedor = Desligado;

Se Temperatura Interna = Muito Baixa e Temperatura Externa = Alta o Aquecedor = Desligado;

Se Temperatura Interna = Muito Baixa e Temperatura Externa = Muito Alta o Aquecedor = Desligado;

Se Temperatura Interna = Baixa e Temperatura Externa <> Muito Alta o Aquecedor = Fraco;

Se Temperatura Interna = Muito Baixa e Temperatura Externa = Normal o Aquecedor = Forte;

Se Temperatura Interna = Muito Baixa e Temperatura Externa = Baixa o Aquecedor = Forte;

Se Temperatura Interna = Muito Baixa e Temperatura Externa = Muito Baixa o Aquecedor = Forte;

Se Temperatura Interna = Alta o Aquecedor = Desligado;

Se Temperatura Interna = Muito Alta o Aquecedor = Desligado.

Regras de Controle do Sistema de Resfriamento

Se Temperatura Interna = Normal o Resfriador = Desligado;

Se Temperatura Interna = Alta e Temperatura Externa = Muito Baixa o Resfriador = Desligado;

Se Temperatura Interna = Muito Alta e Temperatura Externa = Baixa o Resfriador = Desligado;

Se Temperatura Interna = Muito Alta e Temperatura Externa = Muito Baixa o Resfriador = Desligado;

Se Temperatura Interna = Alta e Temperatura Externa <> Muito Baixa o Resfriador = Fraco;

Se Temperatura Interna = Muito Alta e Temperatura Externa = Normal o Resfriador = Forte;

Se Temperatura Interna = Muito Alta e Temperatura Externa = Alta o Resfriador = Forte;

Se Temperatura Interna = Muito Alta e Temperatura Externa = Muito Alta o Resfriador = Forte;

Se Temperatura Interna = Baixo o Resfriador = Desligado;

Se Temperatura Interna = Muito Baixo o Resfriador = Desligado;

Regras de Controle do Sistema de Ventilação

Se o CO₂ = Muito Alto o Ventilador = Forte;

Se Temperatura Interna = Baixa e Temperatura Externa = Muito Alta o Ventilador = Forte;

Se Temperatura Interna = Alta e Temperatura Externa = Muito Baixo o Ventilador = Forte;

Se CO₂ = Alto e Temperatura Interna = Muito Alta o Ventilador = Fraco;

Se CO₂ = Alto e Temperatura Interna = Muito Baixo o Ventilador = Fraco;

Se CO₂ = Alto e Temperatura Interna = Normal o Ventilador = Fraco;

Se CO₂ = Normal e Temperatura Interna = Muito Alta o Ventilador = Desligado;

Se CO₂ = Normal e Temperatura Interna = Muito Baixo o Ventilador = Desligado;

Se CO₂ = Normal e Temperatura Interna = Normal o Ventilador = Desligado;

Regras de Controle do Sistema de Umidificação

Se Umidade Interna = Muito Baixa e Temperatura Interna <> Muito Alta o Umidificador = Forte;

Se Umidade Interna = Muito Baixa e Temperatura Interna = Muito Alta o Umidificador = Fraco;

Se Umidade Interna = Baixa e Temp. Interna = Muito Baixa o Umidificador = Fraco;

Se Umidade Interna = Baixa e Temp. Interna = Baixa o Umidificador = Fraco;

Se Umidade Interna = Baixa e Temp. Interna = Normal o Umidificador = Fraco;

Se Umidade Interna = Normal e Umidade Externa = Muito Baixa e CO₂ = Alto o Umidificador = Fraco;

Se Umidade Interna = Normal e Umidade Externa = Muito Baixa e CO₂ = Muito Alto o Umidificador = Fraco;

Se Umidade Interna = Baixa e Temp. Interna = Alta o Umidificador = Desligado;

Se Umidade Interna = Baixa e Temperatura Interna = Muito Alta o Umidificador = Desligado;

Se Umidade Interna = Alta ou Umidade Interna = Muito Alta o Umidificador = Desligado;

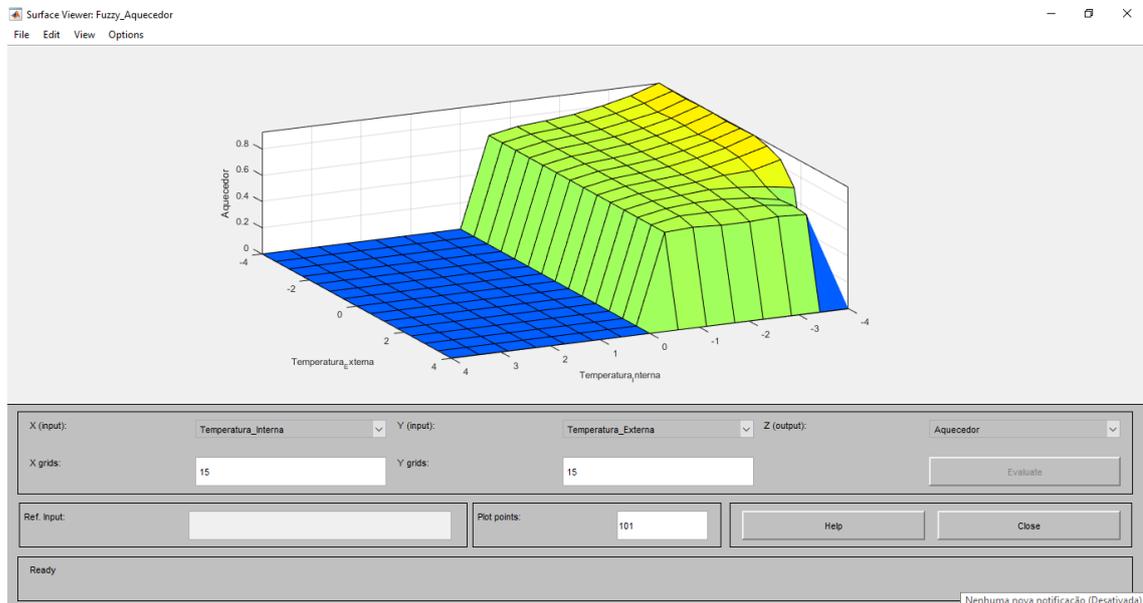
Se Umidade Interna = Normal e Umidade Externa <> Muito Baixa o Umidificador = Desligado;

Se Umidade Interna = Normal e Umidade Externa = Muito Baixa e CO₂ = Normal o Umidificador = Desligado;

APÊNDICE E - SIMULAÇÕES DA POTÊNCIA DE SAÍDA DO CONTROLADOR

O gráfico da potência de saída do aquecedor em função da temperatura interna e temperatura externa é mostrado na figura 43.

Figura 43 – Potência de Saída do Aquecedor.

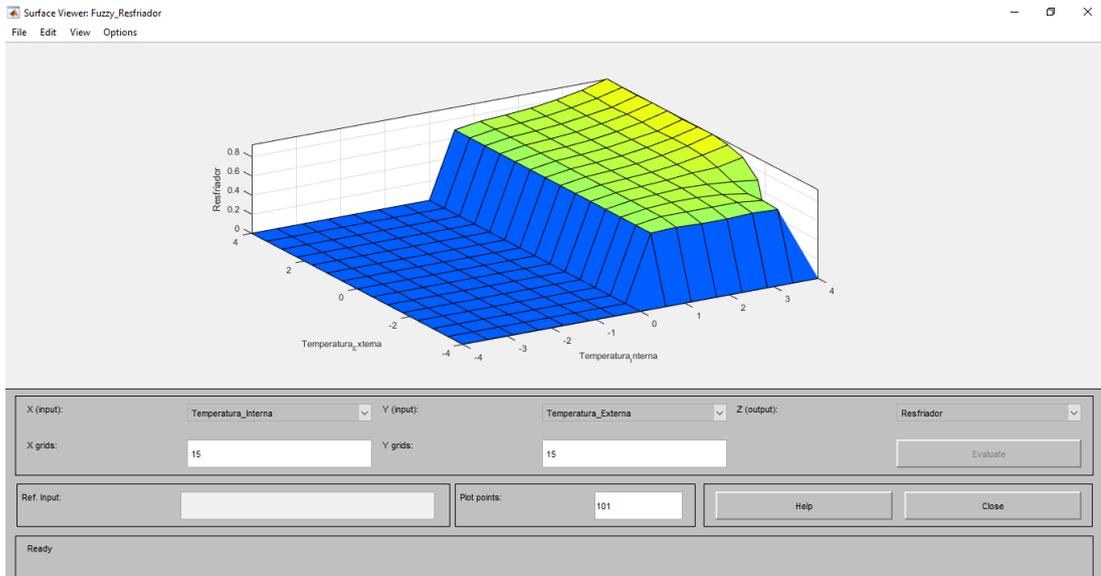


Fonte: 2019, Autor

De acordo com este gráfico é possível observar que a potência de saída é 0 quando a temperatura ultrapassa o set-point. Já quando a temperatura interna medida é reduzida a potência de saída do aquecedor começa a aumentar. É importante observar também que a temperatura externa influencia o controlador somente quando há o extremo em que a temperatura interna está igual ou abaixo do set-point – 4°C e a temperatura externa set-point + 4°C, que neste caso o aquecedor está com a potência zerada, e o aumento da temperatura será feita somente por meio do sistema de ventilação.

O gráfico da potência de saída do resfriador em função da temperatura interna e temperatura externa é mostrado na figura 44.

Figura 44 – Potência de Saída do Resfriador

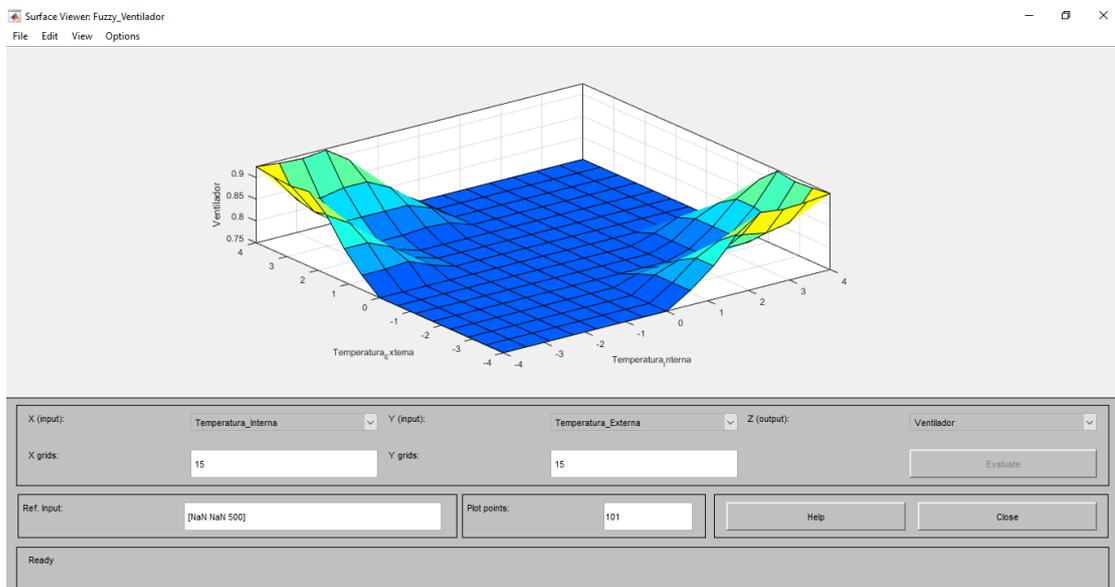


Fonte: 2019, Autor

De acordo com este gráfico é observado que a potência de saída é 0 quando a temperatura está abaixo do *set-point*. Quando a temperatura interna medida é aumentada, a potência de saída do resfriador começa a aumentar. É importante observar também que a temperatura externa influencia o controlador somente quando há o extremo em que a temperatura interna está igual ou acima do *set-point* + 4°C e a temperatura externa *set-point* - 4°C, que neste caso o resfriador está com a potência zerada, e a redução da temperatura será feita somente por meio do sistema de ventilação.

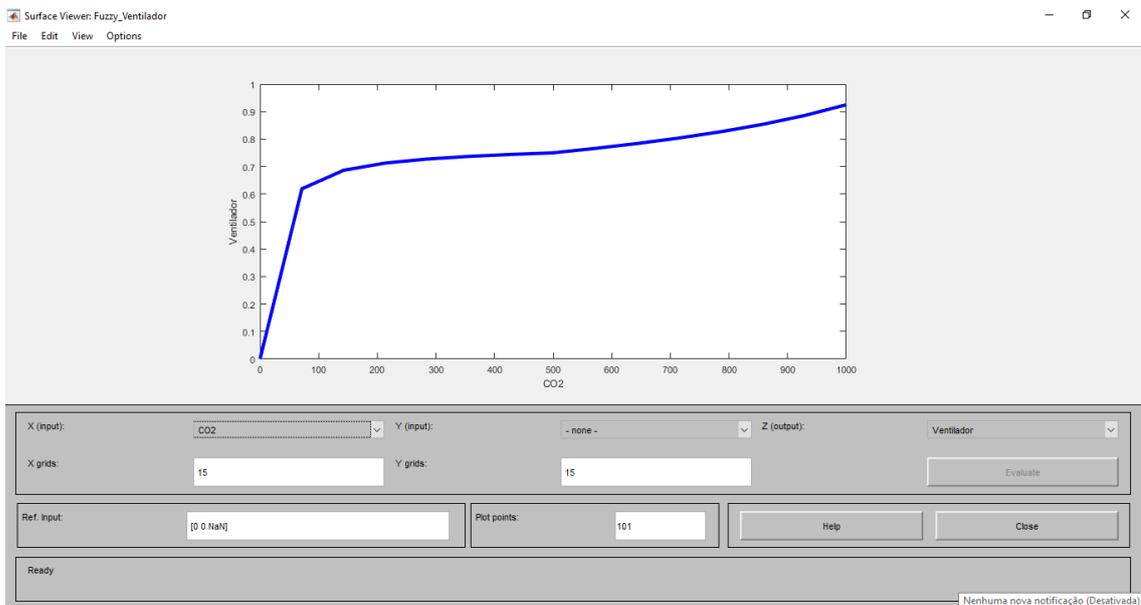
O gráfico da potência de saída do ventilador em função da temperatura interna e temperatura externa é mostrado na figura 45.

Figura 45 – Potência de Saída do Ventilador



De acordo com este gráfico observa-se que a potência de saída do ventilador em função da temperatura interna e externa é maior que zero somente nos extremos. Onde a temperatura interna é muito diferente da externa. Nestas situações o sistema de controle utiliza essa diferença de temperatura para ajustar a temperatura interna com economia de energia, evitando a energização do aquecedor ou resfriador. O gráfico da potência de saída do ventilador em função do nível de CO₂ no interior da estufa de cultivo é mostrado na figura 46.

Figura 46 – Potência de Saída do Ventilador x CO₂



De acordo com este gráfico é possível observar que a potência de saída aumenta em função do aumento do nível de CO₂ medido. O sistema de ventilação tem como função principal o controle do nível de CO₂ no interior da estufa.

APÊNDICE F - CÁLCULO DA PERDA DE CALOR CONTÊINER PARA O AMBIENTE

O modelo matemático para o fluxo de calor do interior da estufa de cultivo para o ambiente externo é dado pela equação 2.

$$\text{VariaçãoTemp.} [^{\circ}C/s] = \frac{(K.A.\Delta T)}{(L.m.c.V)} \quad (2)$$

$K = 0,024$ [W/mK] “Condutividade Térmica da Espuma de Poliuretano”

$L = 0,1$ [m] “Espessura do isolamento térmico”

$A = 114$ [m²] “Área total da parede da estufa + teto”

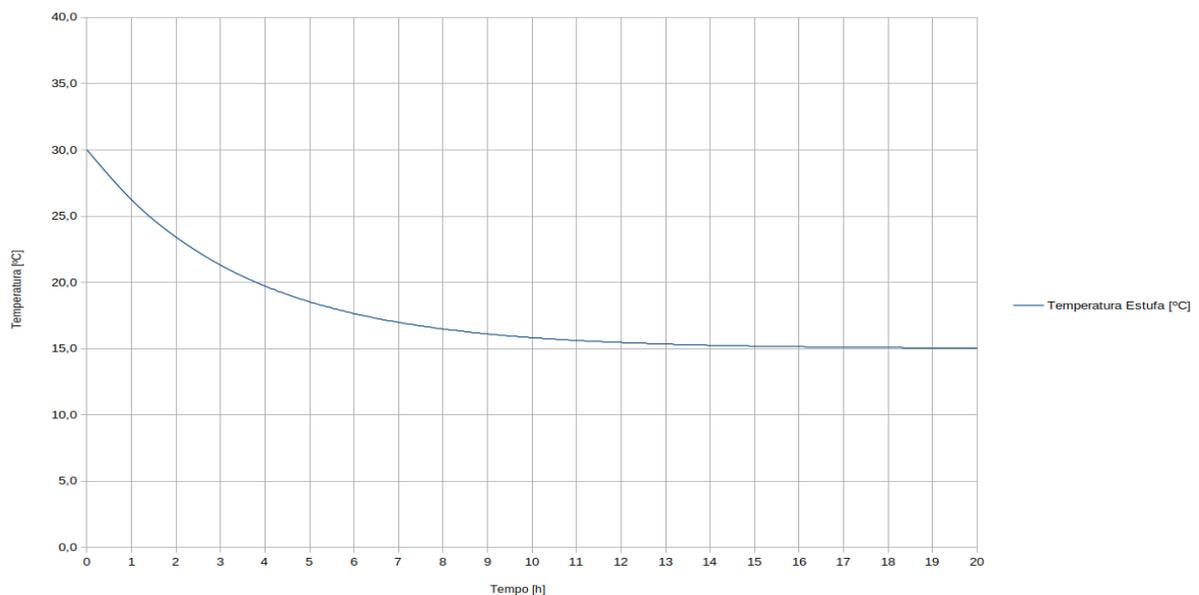
$m = 1,225$ [kg/m³] “Densidade de massa do ar”

$c = 1004$ [J/(kg.K)] “Capacidade térmica do ar”

$V = 76,3$ [m³] “Volume interno da estufa”

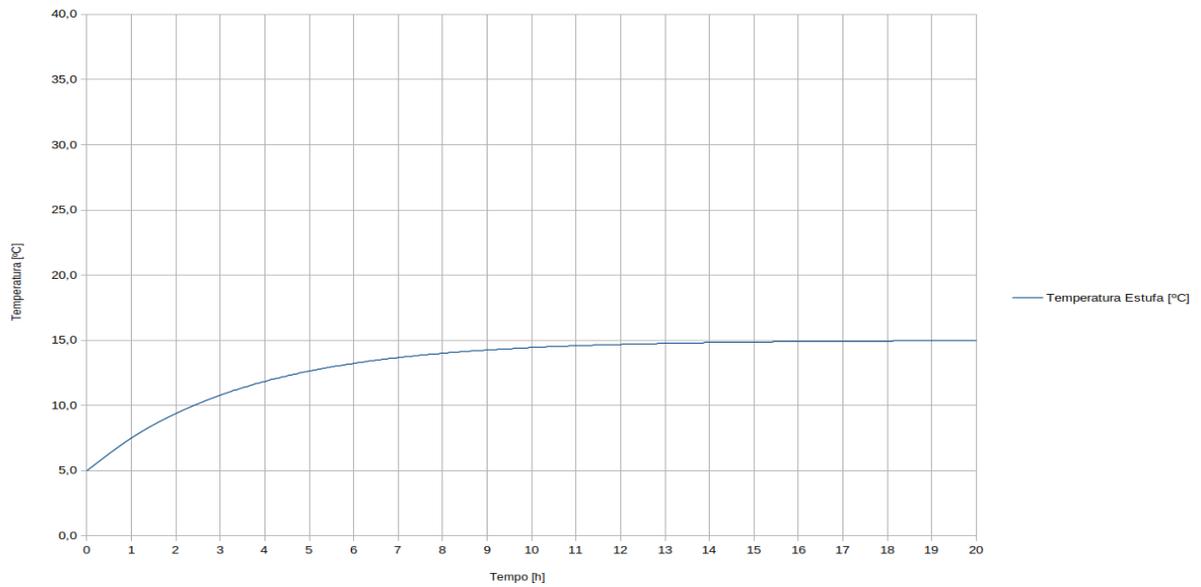
A figura 47, mostra um teste da perda de calor, de uma estufa inicialmente em 30°C, para um ambiente externo com temperatura constante de 15°C, com os parâmetros indicados no capítulo 4.5.1. A figura 48 mostra a situação de uma estufa, inicialmente em 5°C que recebe calor do ambiente que se encontra a uma temperatura constante de 15°C.

Figura 47 – Simulação Transferência de Calor do a Estufa para o Ambiente



Fonte: 2019, Autor

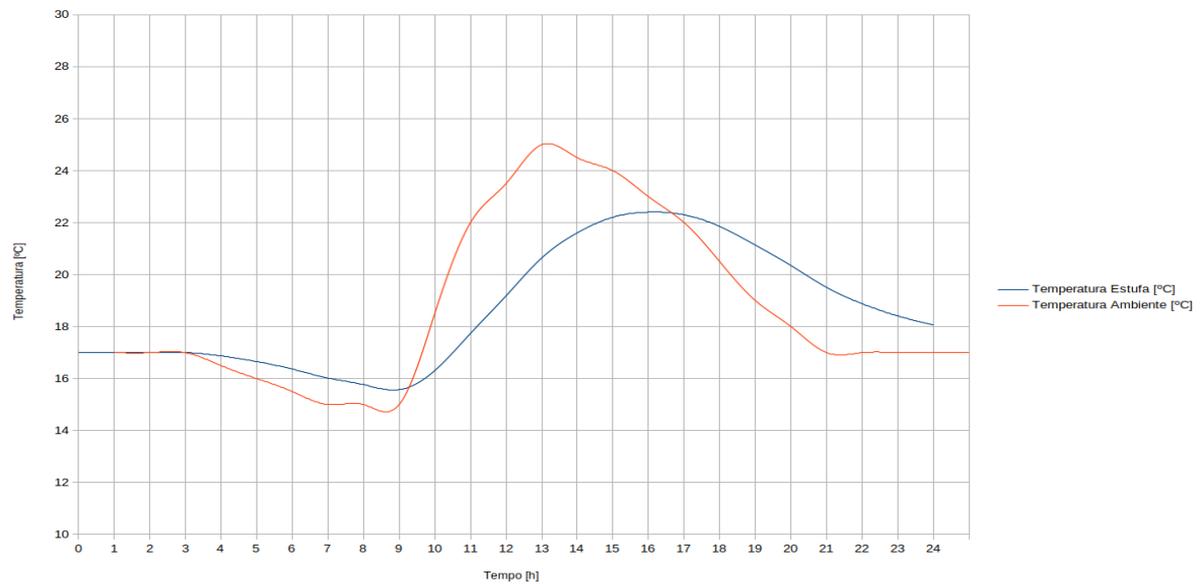
Figura 48 – Simulação Transferência de Calor do Ambiente para a Estufa



Fonte: 2019, Autor

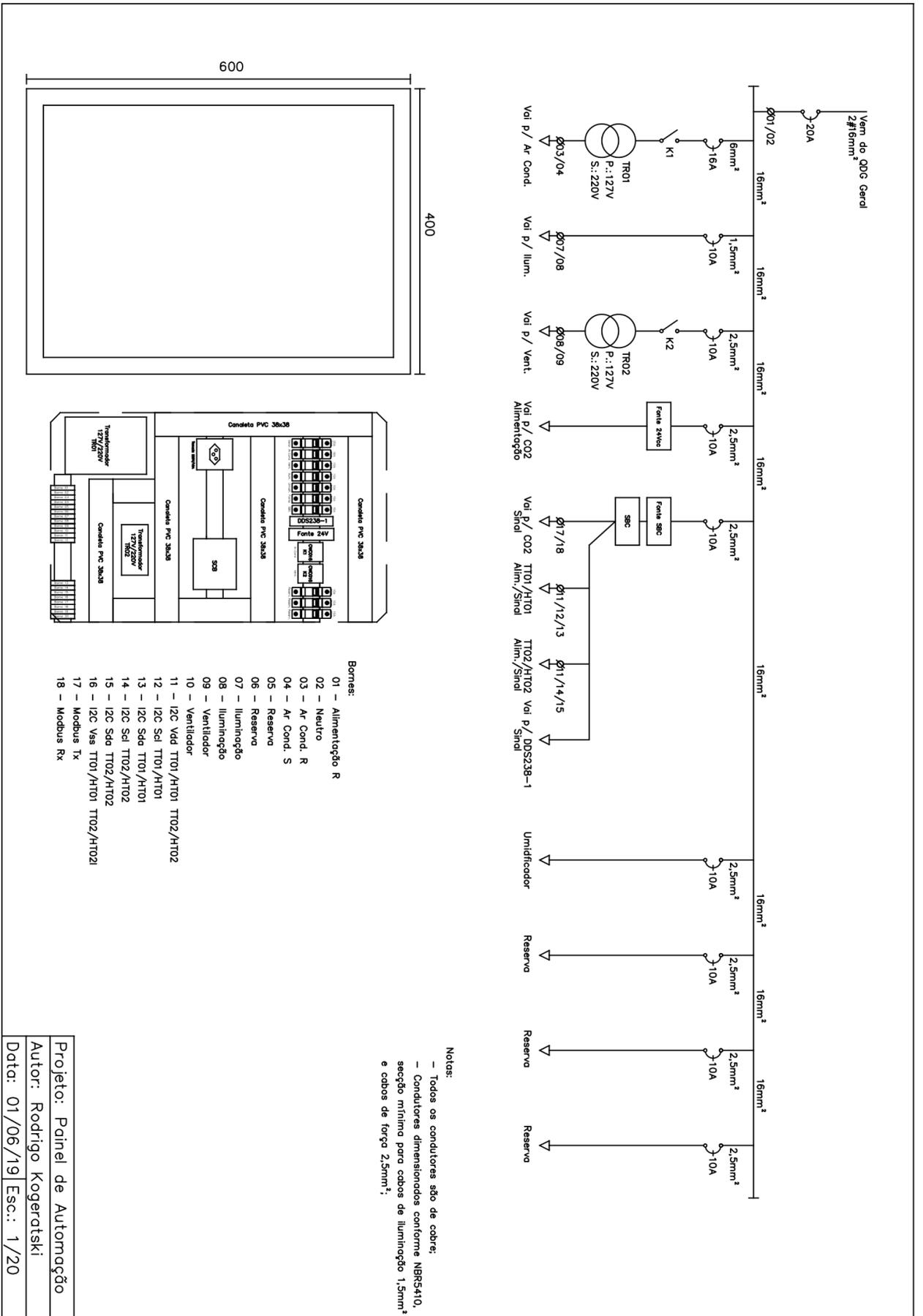
A figura 49 mostra uma simulação para a variação da temperatura no interior da estufa de cultivo sem nenhuma ação de controle de temperatura e com temperatura do ambiente externo variável.

Figura 49 – Simulação Transferência com Temperatura Ambiente Variável



Fonte: 2019, Autor

APÊNDICE G - PROJETO PAINEL DE AUTOMAÇÃO



- Bornes:**
- 01 – Alimentação R
 - 02 – Neutro
 - 03 – Ar Cond. R
 - 04 – Ar Cond. S
 - 05 – Reserva
 - 06 – Reserva
 - 07 – Iluminação
 - 08 – Iluminação
 - 09 – Ventilador
 - 10 – Ventilador
 - 11 – 12C VAd TT01/HT01 TT02/HT02
 - 12 – 12C Sd TT01/HT01
 - 13 – 12C Sda TT01/HT01
 - 14 – 12C Sd TT02/HT02
 - 15 – 12C Sda TT02/HT02
 - 16 – 12C Vas TT01/HT01 TT02/HT02I
 - 17 – Modbus Tx
 - 18 – Modbus Rx

Notas:

- Todos os condutores são de cobre;
- Condutores dimensionados conforme NBR5410, seção mínima para cabos de iluminação 1,5mm² e cabos de força 2,5mm²;

Projeto: Painel de Automação
 Autor: Rodrigo Kogeratski
 Data: 01/06/19 Esc.: 1/20