



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
COLEGIADO DO CURSO DE ENGENHARIA DE COMPUTAÇÃO

Vinícius de Abreu Bozzi

Desenvolvimento e integração de um torrador de café usando Internet das Coisas

Vitória, ES

2022

Vinícius de Abreu Bozzi

Desenvolvimento e integração de um torrador de café usando Internet das Coisas

Monografia apresentada ao Colegiado do Curso de Engenharia de Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Universidade Federal do Espírito Santo – UFES

Centro Tecnológico

Colegiado do Curso de Engenharia de Computação

Orientador: Prof. Vinícius Fernandes Soares Mota

Vitória, ES

2022

Vinícius de Abreu Bozzi

Desenvolvimento e integração de um torrador de café usando Internet das Coisas/ Vinícius de Abreu Bozzi. – Vitória, ES, 2022-
69 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Vinícius Fernandes Soares Mota

Monografia (PG) – Universidade Federal do Espírito Santo – UFES
Centro Tecnológico
Colegiado do Curso de Engenharia de Computação, 2022.

1. Café. 2. Torrefação. 3. Internet das Coisas. 4. Protótipo. 5. Aplicativo Mobile.
6. Android. I. Vinícius de Abreu Bozzi. II. Universidade Federal do Espírito Santo.
IV. Desenvolvimento e integração de um torrador de café usando Internet das Coisas

CDU 02:141:005.7

Vinícius de Abreu Bozzi

Desenvolvimento e integração de um torrador de café usando Internet das Coisas

Monografia apresentada ao Colegiado do Curso de Engenharia de Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. Vitória, ES, 12 de agosto de 2022:

Prof. Vinícius Fernandes Soares Mota
Universidade Federal do Espírito Santo
Orientador

Prof. Giovanni Ventorim Comarela
Universidade Federal do Espírito Santo

**Prof. André Georghthon Cardoso
Pacheco**
Universidade Federal do Espírito Santo

Vitória, ES
2022

Dedico este trabalho à minha família, em especial, ao meu querido irmão Matheus, que esteve nessa dura jornada comigo.

À minha querida mãe Márcia, que sempre me apoiou e deu suporte em tudo que precisei.

Ao meu irmão Fabrício, um espelho para mim.

À minha avó Luzia e tia Mônica que sempre me apoiaram na continuidade dos estudos.

Ao meu pai Sávio, que prestou assistência necessária para minha estadia fora de casa.

À minha querida namorada Eduarda, que me acompanhou em toda essa jornada, sempre me apoiando com muito amor e carinho.

Agradecimentos

Agradeço primeiramente a Deus, por ter me agraciado com saúde durante toda a minha vida.

Ao PET Engenharia de Computação pelos 2 anos de aprendizado.

Ao meu orientador Vinícius Mota. Professor, obrigado pelo seu tempo, sua paciência e por ter aceitado levar adiante esse projeto.

Por fim, agradeço também a todos que contribuíram de forma direta e indireta para a minha formação, não só como engenheiro mas como cidadão.

Obrigado!

*“O segredo do sucesso é a constância do propósito.
(Benjamin Disraeli)”*

Resumo

O café é uma das bebidas mais consumidas no mundo e atualmente vem sendo considerada uma bebida artesanal. Para especialistas, existem muitas etapas para conseguir obter uma boa bebida do café e entre elas está o processo de torrefação. Entretanto, equipamentos de torrefação são caros e muitas vezes inacessíveis e por isso vem sendo difundida a ideia de utilizar equipamentos de baixo custo para auxiliar com este processo, como por exemplo as pipoqueiras elétricas. Neste contexto, o presente trabalho apresenta um sistema para controlar a etapa de torrefação do café. Para o controle desse sistema, foi desenvolvido um aplicativo móvel (Android) que permite aos usuários definir perfis de torras e compartilhá-los com outros usuários. Além disso, a aplicação possui um serviço que permite o acompanhamento do processo de torrefação em tempo real. Para validar a proposta, foi elaborado um protótipo de baixo custo utilizando uma pipoqueira elétrica e considerando-a como um objeto conectado à Internet das Coisas. No protótipo foi utilizado um microcontrolador e sensoriamento, que possibilita a comunicação de dados com a nuvem bem como a realização do controle da temperatura do equipamento (fundamentando-se na teoria de sistemas de controle realimentado). Após o desenvolvimento do trabalho, foi possível fazer as torras de cafés de forma controlada (dentro das limitações da pipoqueira elétrica), isto é, reproduzindo a curva de torra pré-definida pelo usuário no aplicativo e também acompanhar o processo de torrefação em tempo real pelo aplicativo mobile.

Palavras-chaves: Café. Torrefação. Internet das Coisas. Protótipo. Aplicativo Mobile. Android.

Lista de ilustrações

Figura 1 – Blocos básicos da IoT	19
Figura 2 – Arquitetura básica de IoT em três camadas.	20
Figura 3 – Nuvem de Coisas.	22
Figura 4 – Processo a ser controlado.	25
Figura 5 – Controle em malha aberta (sem realimentação).	25
Figura 6 – Controle em malha fechada (com realimentação).	25
Figura 7 – Resposta ao degrau unitário típico de um sistema de controle, ilustrando as especificações no domínio do tempo.	29
Figura 8 – Participação de mercado de sistemas operacionais para celulares e tablets no Brasil. Julho 2021 a Julho 2022	32
Figura 9 – Componentes do projeto na malha de controle - Circuito de Aquecimento	34
Figura 10 – Arquitetura do projeto	35
Figura 11 – Pipoqueira Britânia Pop Time	36
Figura 12 – NodeMCU ESP-WROOM-32	36
Figura 13 – Max6675 e termopar tipo K	37
Figura 14 – Relé eletromecânico Tongling	38
Figura 15 – Solid State Relay - SSR Fotek	38
Figura 16 – Circuito Elétrico - Pipoqueira Britânia Pop Time	39
Figura 17 – Circuito Elétrico (Ventilação e Aquecimento Separados) - Pipoqueira Britânia Pop Time	39
Figura 18 – Circuito Elétrico Completo	40
Figura 19 – Montagem original	41
Figura 20 – Separando os circuitos	41
Figura 21 – Ligação dos fios para o motor DC	41
Figura 22 – Furo de acoplamento do sensor de Temperatura	42
Figura 23 – Passagem dos fios por debaixo da pipoqueira	42
Figura 24 – Circuito completo - montagem	42
Figura 25 – Estética final da montagem	42
Figura 26 – Obtendo API_KEY do Firebase	45
Figura 27 – Obtendo DATABASE_URL do Firebase	45
Figura 28 – Obtendo USER_EMAIL e USER_PASSWORD do Firebase	46
Figura 29 – Configuração de conexão do Aplicativo e Firebase	46
Figura 30 – Diagrama de casos de uso	47
Figura 31 – Tela Inicial Splash	48
Figura 32 – Tela de Login e Cadastro	48
Figura 33 – Tela Principal (a) “Minhas Torras”	48

Figura 34 – Tela Principal (b) “Minhas Torras”	48
Figura 35 – Tela de Adicionar Nova Torra	49
Figura 36 – Tela de Cadastro de Dispositivos - Ler na rede	50
Figura 37 – Tela de Cadastro de Dispositivos - QR Code	50
Figura 38 – Tela Torras Globais	50
Figura 39 – Tela de Torra Seleccionada - Visão Geral	51
Figura 40 – Tela de Torra Seleccionada - Enviar curva de torra	51
Figura 41 – Tela de Torra Seleccionada - Pré-aquecendo	51
Figura 42 – Tela de Torra Seleccionada - Iniciar Torra	52
Figura 43 – Tela de Torra Seleccionada - Fim da Torra	52
Figura 44 – Estrutura do Banco de Dados	53
Figura 45 – Cadastro de novo usuário	55
Figura 46 – Criação de nova torra	55
Figura 47 – Tela principal “Minhas Torras”	55
Figura 48 – Café verde para a torra	56
Figura 49 – Definição das constantes do PID pela entrada Serial	56
Figura 50 – Tela de Torra Seleccionada “torra 1”	57
Figura 51 – Tela de Torra Seleccionada - Fim da Torra - $K_p=5$ $K_i=5$ $K_d=5$	57
Figura 52 – Tela de Torra Seleccionada - Fim da Torra - $K_p=40$ $K_i=1$ e $K_d=50$	58
Figura 53 – Acompanhamento em tempo real - “torra 2”	59
Figura 54 – Café verde	60
Figura 55 – Café após Torrefação - “torra 2”	60
Figura 56 – Acompanhamento em tempo real - “torra 3”	61
Figura 57 – IKAWA Home Roasting System	62

Lista de tabelas

Tabela 1 – Planos de preços - Firebase - Google Cloud	23
Tabela 2 – Características do microcontrolador NodeMCU ESP-WROOM-32 . . .	37
Tabela 3 – Custo Estimado do Protótipo	61

Lista de abreviaturas e siglas

CoAp	<i>Constrained Application Protocol</i>
CoT	<i>Cloud of Things</i>
GHz	Gigahertz
HTTP	Protocolo de Transferência de Hipertexto
IAE	<i>Integral Absolute Error</i>
IDE	<i>Integrated Development Environment</i>
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IoT	<i>Internet of Things</i>
ISE	<i>Integral Square Error</i>
IP	<i>Internet Protocol</i>
JSON	<i>JavaScript Object Notation</i>
LoRa	<i>Long Range</i>
MAN	<i>Metropolitan Area Network</i>
MQTT	Transporte de Telemetria da Fila de Mensagens
M2M	<i>Machine-to-Machine</i>
OSI	<i>Open Systems Interconnection</i>
OTA	<i>Over-the-air</i>
PID	Proporcional Integral Derivativo
RFID	<i>Radio-Frequency Identification</i>
SDK	<i>Software Development Kit</i>
VDC	<i>Voltage Direct Current</i>
TI	Tecnologia da Informação
Wi-Fi	<i>Wireless Fidelity</i>
WAN	<i>Wide Area Network</i>
WLAN	<i>Wireless Local Area Network</i>

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	15
1.2	Motivação e Justificativa	15
1.3	Método de Desenvolvimento do Trabalho	16
1.4	Organização da Monografia	17
2	REFERENCIAL TEÓRICO	18
2.1	Internet das Coisas	18
2.1.1	Definição	18
2.1.2	Arquitetura IoT	20
2.1.3	Protocolos IoT	20
2.1.4	Nuvem de Coisas	22
2.1.5	Desafios e Barreiras	23
2.2	Sistema de Controle Realimentado	24
2.2.1	Definição	25
2.2.2	Os Componentes Básicos de Sistemas de Controle	26
2.2.2.1	Sensores e Transmissores	26
2.2.2.2	Elementos de Controle Final (Atuadores)	26
2.2.2.3	Controladores e Seus Tipos	27
2.2.3	Sintonia de Controle	28
2.2.4	Índices de Desempenho	30
2.3	Dispositivos Móveis e Aplicativos	30
2.3.1	Definição	30
2.3.2	Sistemas Operacionais	31
2.3.2.1	Android	31
2.3.2.1.1	Java	32
2.4	A Técnica Leito Fluidizado (Air Roasting)	33
3	DESENVOLVIMENTO	34
3.1	Desenvolvimento do Hardware	35
3.1.1	Materiais utilizados	35
3.1.1.1	Pipoqueira Elétrica	35
3.1.1.2	Microcontrolador	36
3.1.1.3	Sensor de Temperatura	37
3.1.1.4	Relés Eletromecânico e de Estado Sólido	37
3.1.1.5	Demais materiais necessários	38

3.1.2	Esquemático do Circuito Elétrico	39
3.1.3	Montagem do Circuito Completo	40
3.2	Desenvolvimento do Software	43
3.2.1	Programação do Microcontrolador	43
3.2.1.1	Configuração do Firebase no Microcontrolador	44
3.2.2	Desenvolvimento Mobile	46
3.2.2.1	Login	47
3.2.2.2	Acessar Torras Particulares	48
3.2.2.3	Criar Torras	49
3.2.2.4	Cadastrar Torradeiras	49
3.2.2.5	Acessar Torras Compartilhadas	50
3.2.2.6	Acompanhar Torra em Tempo Real	50
3.2.3	Estrutura do Banco de Dados	52
4	VALIDAÇÃO DA PROPOSTA	54
4.1	Configurações no Aplicativo	54
4.2	Testagem da Torrefação	56
4.2.1	Testagens para Sintonia do Controlador	56
4.2.2	Testagem de Perfil de Torra	59
4.3	Custos Diretos do Protótipo	61
4.4	Comparativo com Equipamentos Similares no Mercado	62
5	CONCLUSÕES	64
5.1	Trabalhos Futuros	65
5.2	Material Complementar	66
	REFERÊNCIAS	67

1 Introdução

As aplicações da Internet das Coisas, do inglês *Internet of Things* (IoT), têm se intensificado a cada dia na vida das pessoas. São grandes os impactos em diversos setores e contextos, como cidades inteligentes, monitoramento remoto, automação residencial, dispositivos inteligentes, entre outras. A ideia principal da IoT refere-se à integração de objetos físicos e virtuais em redes conectadas à Internet, permitindo que diversos dispositivos utilizados no dia-a-dia, nomeados como “coisas”, conectem-se e troquem informações com finalidade de agilizar e facilitar a vida das pessoas. (ALMEIDA, 2015).

Com o avanço da Internet, da computação em nuvem e o baixo custo de microcontroladores e componentes eletrônicos, os dispositivos e equipamentos inteligentes (também conhecidos como dispositivos *smarts*) podem representar uma grande oportunidade para o empreendedorismo. Esses avanços têm potencial para abrir novos mercados e gerar negócios inovadores no país, além de contribuir de forma significativa na popularização de dispositivos inteligentes de baixo custo.

De acordo com Dorf e Bishop (2018), a chegada da era da Internet das Coisas apresenta muitos desafios intrigantes na aplicação de sistemas de controle no ambiente (como o uso mais eficiente de energia em casas e empresas), bens de consumo, energia, dispositivos médicos, transporte, entre muitos outros. Além disso, ainda segundo Dorf e Bishop (2018), a implementação dos próprios sistemas de controle também está se tornando mais automatizada, mais uma vez assistida por muitos recursos prontamente disponíveis na Internet, associados ao acesso a computadores, sensores e atuadores relativamente baratos.

Todos os aspectos citados acima estão em conformidade com o Plano Nacional de Internet das Coisas - IoT, visto que um dos objetivos é “acelerar a implantação da Internet das Coisas como instrumento de desenvolvimento sustentável da sociedade brasileira, capaz de aumentar a competitividade da economia, fortalecer as cadeias produtivas nacionais, e promover a melhoria da qualidade de vida” (BRASIL, 2017).

Neste contexto, o presente trabalho que será apresentado refere-se a uma aplicação IoT baseando-se no desenvolvimento de um aplicativo para dispositivos móveis (Android) que auxilia no processo de torrefação de café. Esse aplicativo possibilita ao usuário configurar, acessar os dados de suas próprias torras e de torras compartilhadas por outros usuários. Pelo aplicativo também é possível criar curvas de torras de café e acompanhá-las em tempo real durante o processo de torrefação. Para isso, foi construído um protótipo de uma torradeira de café de baixo custo, utilizando um microcontrolador e sensoriamento com finalidade de permitir a sua comunicação sem fio à Internet e realizar o controle da

temperatura da torradeira.

1.1 Objetivos

O objetivo deste trabalho consiste no desenvolvimento de um serviço IoT que permitirá auxiliar no processo de torrefação de café. Para isso, um aplicativo de celular foi criado para que, de maneira interativa com o usuário, seja possível realizar a configuração de curvas de torra, bem como compartilhá-las com outros usuários e será possível acompanhá-las em tempo real durante o processo de torrefação.

Além disso, o objetivo consiste em desenvolver um protótipo de um torrador de café de baixo custo, transformando uma pipoqueira elétrica em um torrador de café, utilizando-se dos conceitos de Internet das Coisas e Sistemas de Controle Realimentado. A pipoqueira elétrica adaptada contará com um sensor e um microcontrolador, que viabilizam o controle dos circuitos internos de ventilação e aquecimento da pipoqueira, bem como a comunicação de dados que serão integrados à nuvem.

1.2 Motivação e Justificativa

Desde as décadas pós-segunda guerra mundial, o consumo de café começou a crescer exponencialmente. Embora de baixa qualidade, para os especialistas é um marco conhecido como primeira onda do café. A segunda onda teria surgido como meio de aumentar a qualidade do café, introduzindo os conceitos de cafés especiais e trazendo a popularização do consumo de café expresso em cafeterias. A terceira onda estaria relacionada à percepção do café como um produto artesanal, diferenciado pelos diversos atributos como qualidade, origem, torra, preparo e comercializado de forma mais direta entre os elos da cadeia (ANDRADE, 2015; BORRELL; MATAIX; CARRASCO-GALLEGO, 2015).

Para os amantes do café, sabe-se que o sabor e aroma que caracterizam a bebida café são resultantes da combinação de centenas de compostos químicos produzidos pelas reações que ocorrem durante a torrefação (MELO, 2004). Devido a essas diversas combinações, um equipamento que realize o processo de torrefação de café permite realizar uma satisfação pessoal, além de tornar possível idealizar uma torrefação exclusiva e padronizada, deixando de se preocupar com padrões de torra já estabelecidos pelas indústrias de cafés.

O fato da torrefação não necessitar de apoio tecnológico é comum, uma vez que é possível atingir resultados de torra eficazes de forma manual, utilizando-se de torradores de tambor (MOTT, 2021). Entretanto, nos procedimentos manuais se torna mais difícil atingir o mesmo padrão de torra para todos os processos. Por outro lado, o usuário utilizando-se de um equipamento elétrico que execute a torrefação de acordo com uma curva torra já pré-configurada, consegue atingir um processo de torra mais preciso e padronizado.

O fato de que os padrões de torra de café já são estabelecidos pelas indústrias faz com que a experiência das pessoas - que realizam estudos sobre o café ou que simplesmente apreciam café e gostariam de dedicar tempo a ele - seja limitada. Desta forma, o projeto consiste no desenvolvimento de uma aplicação IoT que conta com um aplicativo móvel e que permite ao usuário criar perfis de torra e compartilhá-las, cadastrar torradeiras e acompanhar o processo da torrefação em tempo real. O projeto também realiza o desenvolvimento de um protótipo de um torrador de café de baixo custo, utilizando-se da ideia principal da Internet das Coisas, com o uso de sensor e microcontrolador, integrados na nuvem. Portanto, este projeto pode auxiliar os amantes de café e/ou pessoas que estudam sobre o café a atingir o processo de torrefação personalizado e padronizado, em um ambiente de torras compartilhadas.

1.3 Método de Desenvolvimento do Trabalho

Primeiramente, para a parte de hardware do projeto, foi desenvolvido, por meio de uma adaptação de uma pipoqueira elétrica, um protótipo de uma torradeira de café, baseando-se no modelo Air Roasting. Na pipoqueira adaptada, foram separados os circuitos de ventilação e de calor para que um microcontrolador pudesse atuar conforme um controlador Proporcional Integral Derivativo (PID). Para esse passo, foi necessário realizar estudos sobre os circuitos elétricos e sistemas realimentados. Como o procedimento envolveu riscos de choques elétricos, foram tomadas medidas de segurança que envolveram evitar o manuseio do equipamento perto de superfícies com água além de realizar os devidos isolamentos elétricos.

Ademais, para a interação com o mecanismo de controle do microcontrolador, foi necessário o estudo sobre comunicação sem fio e seus protocolos. Além dos estudos, foram aplicados conhecimentos de linguagens de programação orientada a objetos para o desenvolvimento de um aplicativo mobile, e de banco de dados de tempo real para o armazenamento de dados gerados. No aplicativo, foram criadas diversas interfaces que permitem ao usuário definir os modelos de torras e acompanhá-las em tempo real, e além disso, o aplicativo permite ou não o compartilhamento de torras entre usuários, configurando-se assim como uma plataforma de torras compartilhadas de multiusuários.

Por fim, foram realizados testes de torrefação e analisados os gastos para o protótipo, vantagens e desvantagens e feita a comparação com produtos como este encontrados no mercado.

1.4 Organização da Monografia

Nesta introdução, foram definidos os objetivos, as motivações e justificativas e o método de desenvolvimento do trabalho. A sequência do trabalho será feita em quatro capítulos que serão divididos da seguinte forma:

- O Capítulo 2 apresenta os conteúdos conceituais utilizados para o desenvolvimento do trabalho. Foi abordado temas como: Internet das Coisas, Sistemas de Controle Realimentados, Dispositivos e Aplicativos Móveis e A Técnica Leito Fluidizado (Air Roasting).
- No Capítulo 3 apresenta as etapas de desenvolvimento do protótipo, compreendidas em duas etapas: hardware e software.
- No Capítulo 4 é feita a validação da proposta, realizando as configurações necessárias e os testes de torrefação. Além disso, são debatidos os custos diretos do protótipo e um comparativo com um equipamento existente no mercado.
- O Capítulo 5 conclui este trabalho citando os pontos positivos e negativos encontrados e também levantando possibilidades de melhorias e ajustes futuros para o protótipo e o aplicativo.

2 Referencial Teórico

O presente capítulo apresenta os referenciais teóricos utilizados para o desenvolvimento deste projeto. São apresentados os conceitos teóricos que guiaram o planejamento do projeto, codificação e validação da proposta. A seção 1.2 aborda os conceitos de Internet das Coisas, pontuando os principais temas utilizados no projeto. A seção 2.2 trata sobre os conceitos de Controle Realimentados, introduzindo os conceitos utilizados para o desenvolvimento e sintonia do controle do protótipo. A seção 2.3 expõe as ideias gerais sobre os dispositivos móveis. Por fim, a seção 2.4 apresenta a técnica *Air Roasting*, usada para fazer torrefações de café.

2.1 Internet das Coisas

2.1.1 Definição

A origem da Internet das Coisas (do inglês *Internet of Things* - IoT) foi atribuída a membros do Auto-ID Center do MIT (*Massachusetts Institute of Technology*), a comunidade de desenvolvimento da *Radio-Frequency Identification* (RFID), por volta de 2000. No princípio, o termo se referia apenas aos objetos etiquetados por RFID. Atualmente, IoT refere-se a um paradigma emergente que consiste no emprego de coisas facilmente legíveis, reconhecíveis, localizáveis, endereçáveis e/ou controláveis que se comunicam para formar uma rede dinâmica mundial (GAMA; TOUSEAU; DONSEZ, 2012).

Segundo Godoi e Araújo (2019), o conceito de Internet das Coisas e suas soluções envolvem diversos campos, como planejamento urbano, produção agrícola, logística, produção industrial, transporte de pessoal, saúde, proteção ambiental, etc. Devido à inserção social da tecnologia, com capacidade de solucionar diversos problemas do cotidiano das pessoas e da vida econômica mundial, a Internet das Coisas tornou-se uma ferramenta indispensável não só para conectar dispositivos inanimados capazes de acessar a Internet, mas que proporcionam uma melhoria na qualidade de vida das pessoas. Portanto, a promoção de serviços avançados por meio da interconexão pode ser entendida como uma infraestrutura global para a era da informação.

Santos et al. (2016) afirmam que a IoT pode ser compreendida como a combinação de várias tecnologias que se complementam ao viabilizar a integração dos objetos no ambiente físico ao mundo virtual. Neste contexto, pode-se definir os blocos básicos de construção da IoT, representada pela Figura 1.

- Identificação: técnicas para identificar os objetos unicamente com finalidade de

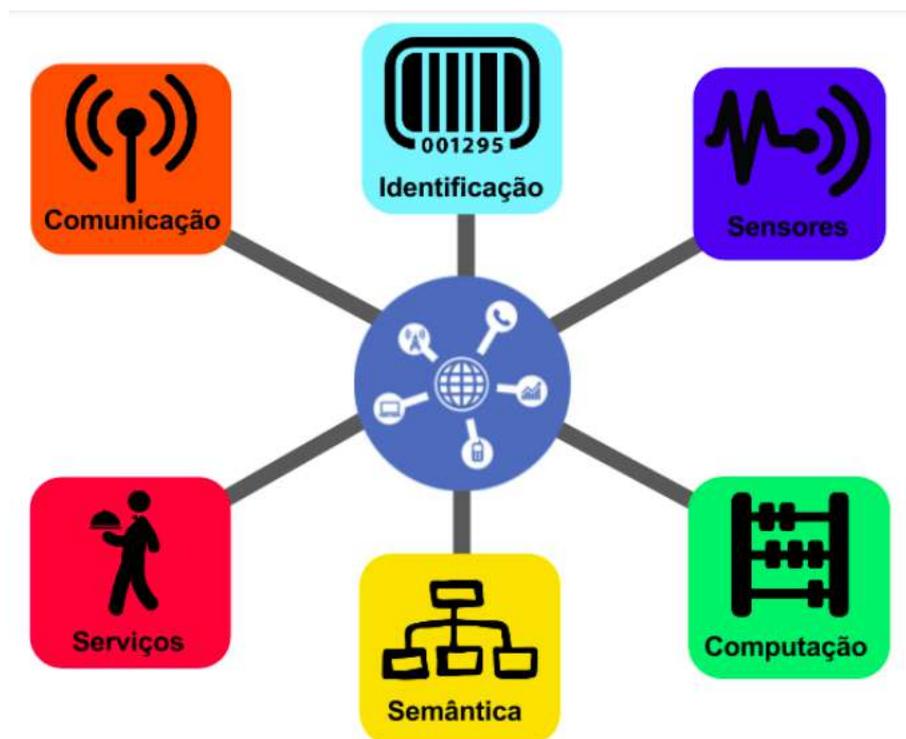


Figura 1 – Blocos básicos da IoT
Fonte: (SANTOS et al., 2016)

conectá-los à Internet. Podem ser empregados para identificar os objetos tecnologias como RFID, NFC (Near Field Communication), endereçamento IP, dentre outras.

- Sensores e Atuadores: coletam informações no contexto em que os objetos se encontram e armazenam os dados em centros de armazenamento, nuvens, etc. Atuadores podem manipular o ambiente ou reagir de acordo com os dados lidos.
- Comunicação: abrange diversas técnicas usadas para conectar objetos inteligentes. Algumas das tecnologias empregadas são WiFi, IEEE 802.15.4, Bluetooth, RFID, LoRa, entre outras.
- Computação: unidade de processamento (microcontroladores, processadores, FPGAs, etc), que executa os algoritmos locais nos objetos inteligentes.
- Serviços: abrange serviços de:
 - Identificação: responsáveis por mapear Entidades Físicas em Entidades Virtuais.
 - Agregação de dados: coletam e sumarizam dados homogêneos ou heterogêneos obtidos dos objetos inteligentes.
 - Colaboração e inteligência: tomam decisões e reagem de modo adequado a um determinado cenário.
 - Ubiquidade: promovem serviços de colaboração e inteligência em qualquer momento ou lugar necessários

- Semântica: trata da descoberta de conhecimento e uso eficiente dos recursos existentes na IoT, a partir dos dados existentes, com o objetivo de prover determinado serviço e em outras palavras, é habilidade de extração de conhecimento dos objetos na IoT.

2.1.2 Arquitetura IoT

Não é uma tarefa simples definir a arquitetura para as aplicações voltadas ao cenário da Internet das Coisas devido à ausência de padrões universais de protocolos de comunicação, bem como a gama enorme de dispositivos físicos utilizados neste contexto (FERRASI; MORGADO, 2018). Na literatura, pode-se encontrar uma variedade de propostas de arquiteturas complexas, baseadas nas necessidades da indústria e academia (AL-FUQAHA et al., 2015). Um modelo mais simples é conhecido como arquitetura de três camadas, ilustrado na Figura 2.

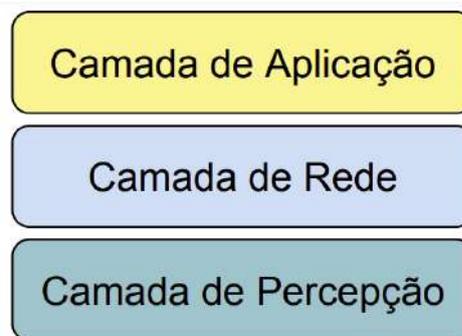


Figura 2 – Arquitetura básica de IoT em três camadas.
Fonte: (SANTOS et al., 2016)

A camada de percepção contém os dispositivos físicos, ou seja, sensores para coletar dados do mundo real, bem como o gateway que fornece os mecanismos e os protocolos para expor os dados detectados à internet (via Wi-Fi, Ethernet, GSM, etc.). Na camada de rede são realizadas as abstrações das tecnologias de comunicação, serviços de gerenciamento, roteamento e identificação. Por fim, a terceira camada é denominada camada de aplicação, uma vez que provê serviços para os clientes, seja por meio de aplicativos móveis, aplicativos da web, dentre outras possibilidades (SANTOS et al., 2016).

2.1.3 Protocolos IoT

Protocolo significa um conjunto de informações, decisões, normas e regras definidas para garantir o funcionamento dos mais variados elementos e serviços disponíveis na rede. Portanto, um conjunto de normas é necessário para possibilitar que máquinas, dispositivos e diversos tipos de software estabeleçam uma negociação ou conversa entre si (MASCHIETTO et al., 2020).

Sob mesma perspectiva, Delicato, Pires e Batista (2013) definem que os protocolos

servem para que dispositivos se comuniquem de forma interoperável e se integrem facilmente a outros dispositivos e sistemas de diferentes fornecedores e desta forma estabelecendo uma língua universal entre todas as marcas, independentemente do fabricante ou modelo utilizado. Os protocolos apresentados a seguir seguem de acordo com as camadas do modelo OSI, que é um modelo de redes de computadores baseado em camadas. Alguns dos mais utilizados em IoT são:

- Camada de aplicação:
 - Protocolo de Transferência de Hipertexto, em inglês *Hypertext Transfer Protocol* (HTTP) - é a base da comunicação de dados para a *World Wide Web*. Hipertexto é um texto estruturado que usa links lógicos (hiperlinks) entre nós que contêm texto. Seu funcionamento segue o princípio de requisição-resposta no modelo computacional cliente-servidor.
 - Transporte de Telemetria da Fila de Mensagens, em inglês *Message Queuing Telemetry Transport* (MQTT) - projetado para ser leve e simples, fundamentado no modelo Publicador-Subscritor. É um dos protocolos mais populares *Machine-to-Machine* (M2M) para enviar fluxos de dados simples de sensores para aplicações.
 - Protocolo de Aplicação Restrita, em inglês *Constrained Application Protocol* (CoAp) - projetado para aplicativos M2M e é especializado na transferência da web para uso com nós restritos e redes restritas em IoT.
- Camada de Rede/Física:
 - Wi-Fi, em inglês *Wireless Fidelity* - baseado no padrão do *Institute of Electrical and Electronics Engineers* (IEEE) 802.11 para dispositivos de rede local sem fio, em inglês *Wireless Local Area Network* (WLAN) e atua na faixa de bandas de frequências de 2,4 GHz e 5 GHz com altas taxas de transferência de dados. Atualmente seu uso é amplo em residências, escritórios, indústrias, etc.
 - LoraWAN - padrão voltado para aplicativos de rede de área ampla, em inglês *Wide Area Network* (WAN) e projetado para redes de baixa potência com comunicação bidirecional, segura e móvel, de baixo custo em IoT.
 - Ethernet - padrão totalmente caracterizado por conectividade com fio em redes locais, em inglês *Local Area Network* (LAN), redes de área metropolitanas, *Metropolitan Area Network* (MAN) e redes de longa distâncias (WAN), é baseado no padrão IEEE 802.3.
 - IPv4 e IPv6 - Diante da escassez do IPv4 (32 bits de endereços disponíveis), houve a necessidade de transição para o IPV6 (128 bits de endereços disponíveis) para a interconexão de objetos, uma vez que estima-se bilhões de novos sensores

necessitando de endereçamento IP, em inglês *Internet Protocol*. O uso de IPv6 proporcionará novas e intermináveis faixas de IPs, além de oferecer um melhor controle e gerenciamento das redes com recursos de auto configuração e níveis de segurança.

2.1.4 Nuvem de Coisas

Enquanto dispositivos IoT produzem uma quantidade de dados enorme, a computação em nuvem fornece mecanismos para lidar com essas limitações de armazenamento e processamento dos dados. A combinação das duas tecnologias é conhecida como Cloud of Things (CoT, ou Nuvem de Coisas) (MASCHIETTO et al., 2020).

Segundo Maschietto et al. (2020), os projetos IoT podem ser voltados a soluções para infraestrutura em nuvem juntamente com soluções embarcadas. Essas soluções embarcadas envolvem sensores, conexões de dispositivos, hardware embarcado, atuador, entre outros. Portanto é imprescindível a integração das soluções embarcadas com os protocolos de comunicação, transmissão/armazenamento de dados em nuvem, aplicativos de controles/acesso, etc. A Figura 3 ilustra a ideia citada.

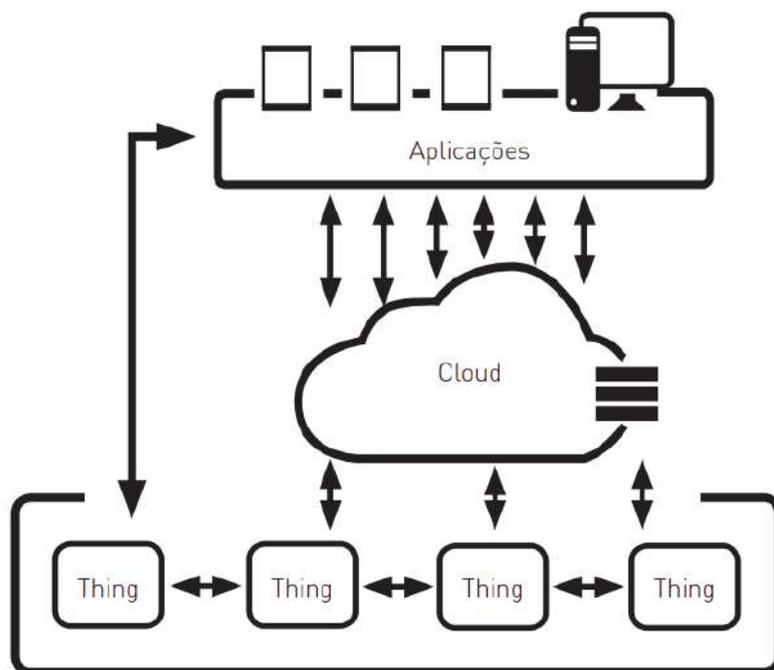


Figura 3 – Nuvem de Coisas.
Fonte: (MASCHIETTO et al., 2020)

Para Maschietto et al. (2020), muitos são os desafios para lidar com uma imensa quantidade de dados gerados. Esses desafios estão relacionados as limitações de armazenamento, de processamento e de comunicação. Também existem diversos desafios referentes a conectividade de dispositivos em tempo real. No entanto, essas questões estão sendo atacadas com a crescente evolução de tecnologias de banda larga, de métodos de acesso

e dos diversos protocolos (que são utilizados para convergência das comunicações entre dispositivos).

Neste contexto, para Borgia (2014), a nuvem pode sanar vários problemas relacionados a IoT de forma eficaz. Problemas como o de processamento de dados e extração de informações úteis são resolvidas como por exemplo, alocando/desalocando processos dinamicamente e aproveitando soluções específicas do contexto de Big Data, respectivamente.

Um dos serviços em nuvem bastante conhecido no mundo IoT é o Google Cloud, que provê diversos tipos de recursos de computação em nuvem utilizados para desenvolver, implantar e operar aplicativos na web. A plataforma é uma infraestrutura completa que conta com mais de 100 produtos e serviços para novas possibilidades e soluções empresariais (TRAFANIUC, 2021). Algumas delas são:

- Máquinas virtuais;
- Hospedagem de sites;
- Armazenamento de arquivos;
- Compartilhamento e processamento de dados; e
- Banco de dados.

O Firebase Realtime Database que é um desses banco de dados do Google Cloud. Segundo o Google (2022), ele é um banco de dados não relacional (NoSQL) hospedado na nuvem que permite armazenar e sincronizar dados entre seus usuários em tempo real. Os dados são armazenados como JSON e sincronizados em tempo real para cada cliente conectado.

O Firebase Realtime Database oferece operações de leitura e gravação de forma gratuitas e ilimitadas. Outros serviços são pagos (plano Blaze) e gratuitos (plano Spark). Esses planos são mostrados na tabela abaixo (GOOGLE, 2022).

Tabela 1 – Planos de preços - Firebase - Google Cloud

Realtime Database	Spark	Blaze
Conexões Simultaneas	100	200 Mil/Banco de Dados
GB de Armazenamento	1 GB	\$5/GB
GB de Download	10 GB/mês	\$1/GB
Bancos de Dados por Projeto	1	Vários

2.1.5 Desafios e Barreiras

Segundo o McKinsey Global Institute (2015), algumas condições devem ser atingidas, e diversos obstáculos devem ser superados para que a IoT atinja o seu máximo impacto

econômico, estimada em US\$ 11.1 trilhões. Primeiramente, a pesquisa defende que alguns desses fatores são estruturais e comportamentais, como as empresas que precisam acreditar e se alinhar com o foco de realizar suas decisões em dados que a IoT fornece. Além disso, foram destacados alguns desafios:

- **Tecnologia:** para a adoção generalizada da Internet das Coisas, o preço do hardware precisa continuar caindo, assim como o custo de computação e armazenamento de dados. Com isso, será necessário mais desenvolvimento de análise e visualização de dados.
- **Interoperabilidade:** para que os dispositivos e os sistemas trabalhem juntos, de forma harmônica, pode ser alcançada adotando padrões abertos e também implementando sistemas ou plataformas que permitem que diferentes sistemas IoT se comuniquem entre si.
- **Privacidade e confidencialidade:** a quantidade enorme de dados gerados por dispositivos IoT precisarão garantir privacidade com transparência para os indivíduos bem como confidencialidade e integridade dos dados para as organizações.
- **Segurança:** com novos dispositivos cria-se mais oportunidades para possíveis violações, que devem ser gerenciados para garantir segurança para acessos não autorizados, consequentemente evitando possíveis danos físicos.
- **Propriedade intelectual:** um entendimento comum dos direitos de propriedade dos dados produzidos por vários dispositivos conectados será necessário para liberar todo o potencial da IoT. A questão é quem terá os direitos e propriedades em relação aos dados coletados. O usuário? O fabricante do produto inteligente? Quem está controlando?
- **Organização e talento:** grande capacidade de adaptação das organizações será requerida com finalidade de adoção destes novos processos e novos modelos de negócio. O setor de tecnologia de informações (TI) terá maior relevância nas organizações, contudo dependerá de um bom alinhamento com outros setores.
- **Políticas públicas:** as aplicações de IoT devem ser regulamentadas para garantir políticas de definição de regras de mercado, de prática de dados, segurança e responsabilidades, bem como incentivos para diversas áreas da sociedade.

2.2 Sistema de Controle Realimentado

Na presente seção, serão abordados os conceitos teóricos de sistemas de controle para um melhor entendimento de sua utilização prática no projeto.

2.2.1 Definição

Segundo Dorf e Bishop (2018), um sistema de controle é uma interconexão de componentes formando uma configuração de sistema que proporcionará uma resposta desejada do sistema. A relação entrada-saída representa a relação de causa e efeito do processo, o qual por sua vez representa o processamento do sinal de entrada para fornecer um sinal de saída desejada.

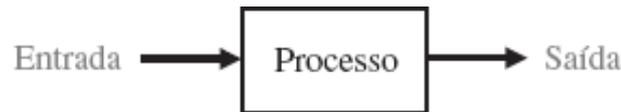


Figura 4 – Processo a ser controlado.

Fonte: (DORF; BISHOP, 2018)

Pode-se definir dois tipos simples de sistemas de controle. O primeiro é um sistema de controle em malha aberta (sem realimentação), que consiste no uso de um controlador e um atuador para obter a resposta desejada, representado pela Figura 5.

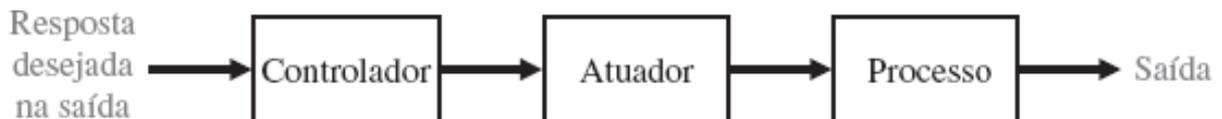


Figura 5 – Controle em malha aberta (sem realimentação).

Fonte: (DORF; BISHOP, 2018)

O segundo é conhecido como sistema de controle de malha fechada (com realimentação), representado pela Figura 6, que implementa uma realimentação com finalidade de compará-la com a resposta desejada, também conhecido como *setpoint*. A diferença entre o *setpoint* e a saída real (medida pelo sensor) resulta no erro do processo, o qual é então corrigido pelo controlador. A saída do controlador faz com que o atuador ajuste o processo para reduzir o erro e assim sucessivamente (DORF; BISHOP, 2018).

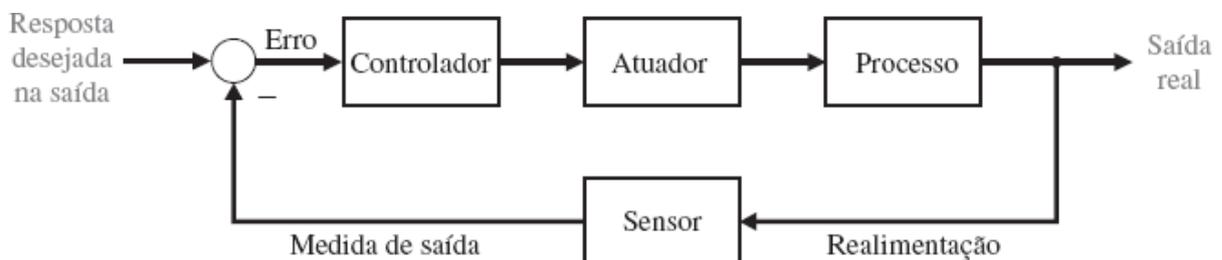


Figura 6 – Controle em malha fechada (com realimentação).

Fonte: (DORF; BISHOP, 2018)

Além disso, segundo Smith e Corripio (2012), para os sistemas de controle, existem três operações básicas que devem estar presentes em qualquer sistema de controle, sendo elas:

1. Medição. A medição da variável a ser controlada é normalmente feita através de sensores.
2. Decisão. Com base na medição, o controlador decide o que fazer para manter a variável em seu valor desejado.
3. Ação. Como resultado da decisão do controlador, o sistema deve então efetuar uma ação, que normalmente é realizada pelo atuador.

Ainda segundo Smith e Corripio (2012), existem os chamados controles regulatórios e servos. O controle regulador se aplica a sistemas desenvolvidos para compensar distúrbios, ou seja, em alguns processos, a saída real desvia do *setpoint* por causa de perturbações externas que afetam o sistema de controle. Já para o controle servo, o distúrbio mais importante é o próprio *setpoint*. Isto é, o *setpoint* pode ser alterado para uma função de tempo, e por esta razão a variável controlada deve seguir o *setpoint*. O controle servo se aplica a sistemas de controle planejados para este propósito, como por exemplo um reator de fornalha, onde a temperatura deve seguir um perfil desejado.

2.2.2 Os Componentes Básicos de Sistemas de Controle

Segundo Smith e Corripio (2012), os componentes básicos de sistemas de controle são os sensores e transmissores, elementos de controle final, também conhecidos como atuadores, e os controladores. Estes componentes são os responsáveis por executar as três operações básicas de sistemas de controle, vistas anteriormente em 2.2.1.

2.2.2.1 Sensores e Transmissores

O sensor produz um fenômeno, seja ele mecânico, elétrico ou semelhante, relacionado à variável do processo que ele mede. O transmissor, por sua vez, tem a função de converter este fenômeno em um sinal que pode ser transmitido (SMITH; CORRIPIO, 2012). Em outras palavras, sensores são dispositivos que têm a função de detectar e responder com eficiência algum estímulo. Ao receber um estímulo, é emitido algum tipo de sinal que seja capaz de ser interpretado por outros dispositivos. São encontrados uma infinidade de sensores que respondem a estímulos diferentes como por exemplo: calor, pressão, movimento, etc (MATTEDE, 2018).

2.2.2.2 Elementos de Controle Final (Atuadores)

São os elementos que realizam a função ação do sistema de controle através do ajuste que afetam diretamente as variáveis a serem controladas (SMITH; CORRIPIO, 2012). Em outras palavras, um atuador é um dispositivo acionado pelo sistema de controle para alterar ou ajustar o ambiente. Um motor elétrico usado para rotacionar um manipulador robótico

é um exemplo de um dispositivo que transforma energia elétrica em torque mecânico (DORF; BISHOP, 2018).

2.2.2.3 Controladores e Seus Tipos

Smith e Corripio (2012) afirmam que existem diversos tipos de controladores conceituados na teoria de controle. Dentre esses, os controladores clássicos e os de lógica nebulosa se destacam por possuir diferentes interpretações, benefícios e custos computacionais. O uso controladores clássicos são muitas vezes utilizados devido sua facilidade de aplicação, baixa demanda computacional e robustez. Esses controladores clássicos e suas combinações são representados pelas constantes proporcional, integral e derivativa.

Para simplificar a apresentação que se segue, utilizou-se todos os sinais na representação de tempo contínuo, uma vez que para controles baseados em computador, usam-se técnicas discretas com formulações matemáticas discretizadas. Entretanto, conforme o tempo de amostragem para estes sistemas é muito rápido, foi simplificada a ideia geral baseando-se no tempo contínuo (SMITH; CORRIPIO, 2012).

Ainda segundo Smith e Corripio (2012), o controlador é o “cérebro” da malha de controle. É ele o responsável pela operação de decisão no sistema de controle e para isso, o controlador:

1. Compara o sinal de processo que ele recebe da variável controlada, com o valor de saída desejado. Este sinal, conhecido como Erro, pode ser representado por:

$$E(t) = R(t) - M(t) \quad (2.1)$$

No qual:

$E(t)$ é o sinal de erro, ou sinal de entrada do controlador;

$R(t)$ é a resposta desejada na saída, ou *setpoint*; e

$M(t)$ é o sinal medido pelo sensor na saída do processo.

2. Envia um sinal adequado para o dispositivo de controle final, para que se mantenha a variável controlada em seu valor de saída desejado. Sinal este representado por $U(t)$.

Dentre os controladores clássicos, o controle proporcional é um tipo de controle representado por um amplificador simples de ganho constante K_p conhecido como constante de proporcionalidade, uma vez que o sinal de controle na saída do controlador ($U_P(t)$) atua de forma proporcional ao erro da realimentação. Com um ganho K_p alto, é gerado um alto sinal de saída, podendo desestabilizar o sistema. Por outro lado, se o K_p é muito

baixo, o sistema pode falhar ao aplicar a ação necessária para corrigir o sinal de controle. O controle proporcional é representado pela equação:

$$U_P(t) = K_p * E(t) \quad (2.2)$$

De acordo com Smith e Corripio (2012), a principal vantagem do controle proporcional é a utilização de um parâmetro para o controle. Entretanto, a desvantagem é operar com um erro residual, isto é, apesar de ser possível estabilizar o sistema, ainda há um erro em relação ao *setpoint* definido. Como a maioria dos processos não admitem esse erro residual, deve-se adicionar ao controlador proporcional, para remover o erro residual um novo modo de controle, conhecido como ação integral ou de restauração, representado por K_i . A ação integral pode acelerar a resposta do sistema, assim podendo chegar ao valor de referência mais rapidamente. Para o K_i baixo, o sistema pode levar mais tempo para atingir o valor de referência. Com K_i alto, o sistema age mais rapidamente porém pode-se tornar muito oscilatório e até se tornar instável. Desta forma, combinando os ganhos citados, o controlador se torna um controlador proporcional-integral (PI). A equação que o descreve é:

$$U_{PI}(t) = K_p * E(t) + K_i * \int_0^t E(\tau) d\tau \quad (2.3)$$

Outra nova constante de controle é adicionada ao controle PI. Este novo modo de controle é a ação derivativa K_d , também conhecida como ação de proporção, ou prévia, pois dá ao controlador a capacidade de prever para onde o processo está caminhando com base no cálculo da derivada do erro. Em outras palavras, o ganho K_d pode antecipar a correção do erro e melhorar a estabilidade do sistema, entretanto o tempo de resposta diminui. Com isso, ao adicionar as três constantes citadas, obtém-se um controlador proporcional-integral-derivativo. A equação descritiva do PID é:

$$U_{PID}(t) = K_p * E(t) + K_i * \int_0^t E(\tau) d\tau + K_d * \frac{dE(t)}{dt} \quad (2.4)$$

2.2.3 Sintonia de Controle

No geral, Golnaraghi e Kuo (2012) afirmam que para a sintonia de controle, as especificações de projeto que devem ser seguidas são únicas para cada aplicação individual e, constantemente, incluem algumas especificações no domínio do tempo, como:

- Estabilidade de um sistema: é quando não há perda do controle do sistema, em outras palavras, para todo sinal de amplitude limitada aplicado em sua entrada, o sinal de saída é também limitado.

- Acurácia no regime estacionário: corresponde ao erro em relação ao *setpoint* no regime estacionário.
- Sobrevalor ou sobressinal máximo: corresponde a um valor máximo medido quando a resposta de um sistema ultrapassa o valor de *setpoint*.
- Tempo de subida: é o tempo relacionado com a velocidade de resposta do sistema.

De maneira geral, para obter bons resultados busca-se um sistema estável e com erro em regime estacionário igual a zero. Além disso, geralmente é indesejável sistemas com sobrevalor alto e desejável sistemas com rápido tempo de subida e conseqüentemente mais velozes (GOLNARAGHI; KUO, 2012). A Figura 7 ilustra essas especificações para uma entrada representada por um sinal degrau.

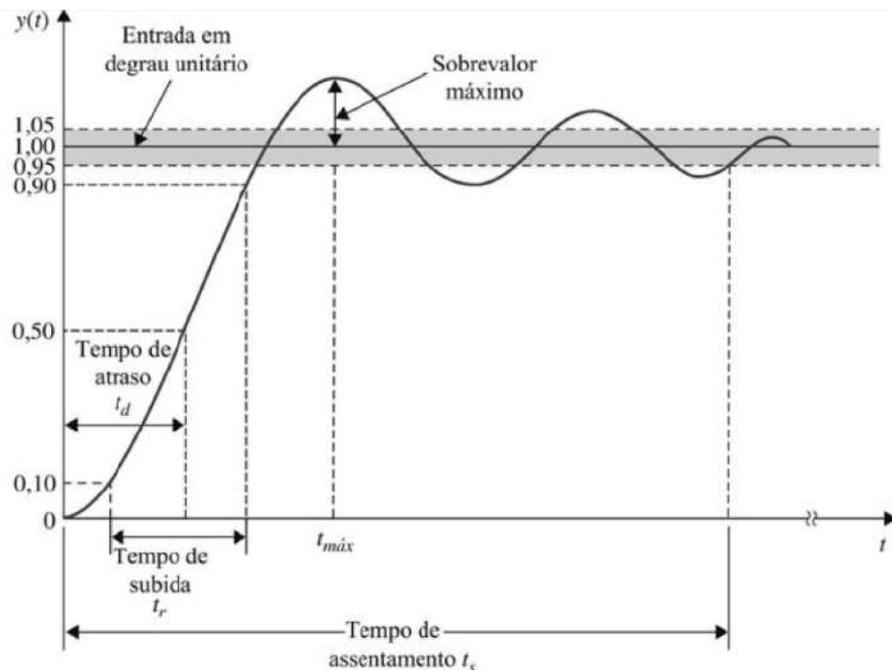


Figura 7 – Resposta ao degrau unitário típico de um sistema de controle, ilustrando as especificações no domínio do tempo.

Fonte: (GOLNARAGHI; KUO, 2012)

Ademais, o objetivo de um controle automático de processo é agir no atuador para manter a variável a ser controlada no setpoint definido, independentemente de perturbações externas. Para atingir esse objetivo, existem muitas técnicas disponíveis para determinar valores aceitáveis das constantes do PID, conhecidos como métodos sintonias de controle PID. Uma dessas técnicas mais comum para a sintonia de controle PID é o uso de métodos de sintonia manual, que consiste em definir as constantes do controlador PID por tentativa e erro com mínima análise analítica usando testes reais nos sistemas e decidindo-se os ganhos baseando-se em observações e experiência (DORF; BISHOP, 2018).

Uma conduta para a sintonia manual é primeiro desligar as ações integral e derivativa do controlador PID de realimentação, ou seja, fazer $K_i = 0$ e $K_d = 0$. A seguir

a constante K_p é incrementada aos poucos até que a saída do sistema em malha fechada oscile exatamente no limiar da estabilidade. Após atingir este ponto, deve-se reduzir o valor de K_p para se atingir o que é conhecido como o decaimento de um quarto da amplitude. Os próximos passos consistem em aumentar K_i e K_d manualmente para alcançar uma resposta desejada. Este método de sintonia é conhecido como sintonia de Ziegler-Nichols, proposto em 1942 por Ziegler e Nichols (SMITH; CORRIPIO, 2012).

2.2.4 Índices de Desempenho

Segundo Dorf e Bishop (2018), na teoria de controle existem índices de desempenho que podem ser calculados ou medidos para avaliar o desempenho do sistema. Essas medidas quantitativas de desempenho são bastante valiosas no projeto e na operação de sistemas de controle. Para que um sistema seja considerado ótimo, os parâmetros do sistema são ajustados de modo que o índice alcance um extremo, geralmente um valor mínimo.

Existem alguns índices de desempenho bastante comuns na teoria de controle. Dentre eles, os dois principais são:

- Integral Absoluta do Erro, em inglês *Integral of Absolute Error* - IAE: é um índice que mede o erro absoluto do sistema, ou seja, é a diferença absoluta entre a resposta desejada e o sinal medido da saída do processo. Pode ser representada pela seguinte equação:

$$IAE = \int_0^T |E(t)| dt \quad (2.5)$$

- Integral Quadrática do Erro, em inglês *Integral of the Square of the Error* - ISE: é um índice que mede o erro quadrático do sistema e desta forma penaliza com grande intensidade os erros maiores em relação à erros menores. É representado pela equação a seguir:

$$ISE = \int_0^T E^2(t) dt \quad (2.6)$$

2.3 Dispositivos Móveis e Aplicativos

2.3.1 Definição

Dispositivos móveis remetem ao conceito de mobilidade, que segundo Lee, Schneider e Schell (2005) é definida como a capacidade de poder se deslocar ou ser deslocado de forma fácil. Para o contexto da computação móvel, esse conceito refere-se ao uso de dispositivos móveis por pessoas. Os dispositivos móveis oferecem um conjunto de aplicações, capazes de conectar-se, obter dados e fornecer esses dados a aplicações, sistemas e a outros usuários.

Ademais, para Lee, Schneider e Schell (2005), um dispositivo móvel possui algumas características como: portabilidade, em que o usuário deve transportá-lo com facilidade;

usabilidade, consiste em ser utilizável por diversos tipos de pessoas em diferentes ambientes; funcionalidade, consiste em servir para múltiplos propósitos e ter diferentes tipos de funcionalidades; e conectividade, consiste em ser possível operar e em conjunto e se comunicar com outros dispositivos.

Lee, Schneider e Schell (2005) destacam em sua obra que os smartphones são os dispositivos mais populares na contemporaneidade, devido às suas características como tela de alta resolução touchscreen que permite uma fácil experiência de visualização, a conectividade presente nestes dispositivos, como a rede Wi-Fi e as conexões de telefonia móvel (GSM, 3G, 4G, 5G, bluetooth, etc), a capacidade de executar aplicativos uma vez que o usuário pode baixá-los em lojas on-line e instalá-los, aumentando as possibilidades de uso para o smartphone, e as câmeras que permitem aos usuários realizar fotos e vídeos em resolução alta.

Um aplicativo nada mais é do que um programa de computador para smartphones, e tal como para os computadores, é necessário um ambiente de execução que são os sistemas operacionais móveis (SIMAS; BORGES; COUTO, 2019).

2.3.2 Sistemas Operacionais

Atualmente, há dois sistemas operacionais que predominam no mercado: o Android, desenvolvido pela Google, utilizado por diversos fabricantes (Samsung, Motorola, LG, Xiaomi, etc.); e o iOS, criado pela Apple e exclusivo para seus dispositivos (SIMAS; BORGES; COUTO, 2019). A Figura 8 ilustra essa presença no Brasil.

Cada um desses sistemas tem uma ou mais linguagens de programação padrão para sua plataforma, que incluem: Java e, mais recentemente, Kotlin, para Android e Objective-C e Swift para iOS.

2.3.2.1 Android

Android é um sistema operacional de código aberto, baseado no núcleo Linux. O projeto começou pela Android, Inc. em 2005 até ser comprada pelo Google em 2005. No mesmo ano um consórcio de desenvolvedores denominado Open Handset Alliance juntamente com o Google desenvolvem e mantêm o projeto (DEITEL; DEITEL; DEITEL, 2015).

Segundo Deitel, Deitel e Deitel (2015), devido a sua grande predominância no mercado de dispositivos móveis, também cresce as oportunidades para os desenvolvedores de aplicativos Android. Existe também grande concorrência entre as fabricantes de dispositivos Android e isso promove a inovação de hardware e software dentro desta comunidade, permitindo ao desenvolvedor Android realizar aplicações cada vez mais avançadas e inovadoras, proporcionando a criação de múltiplas oportunidades neste mercado.

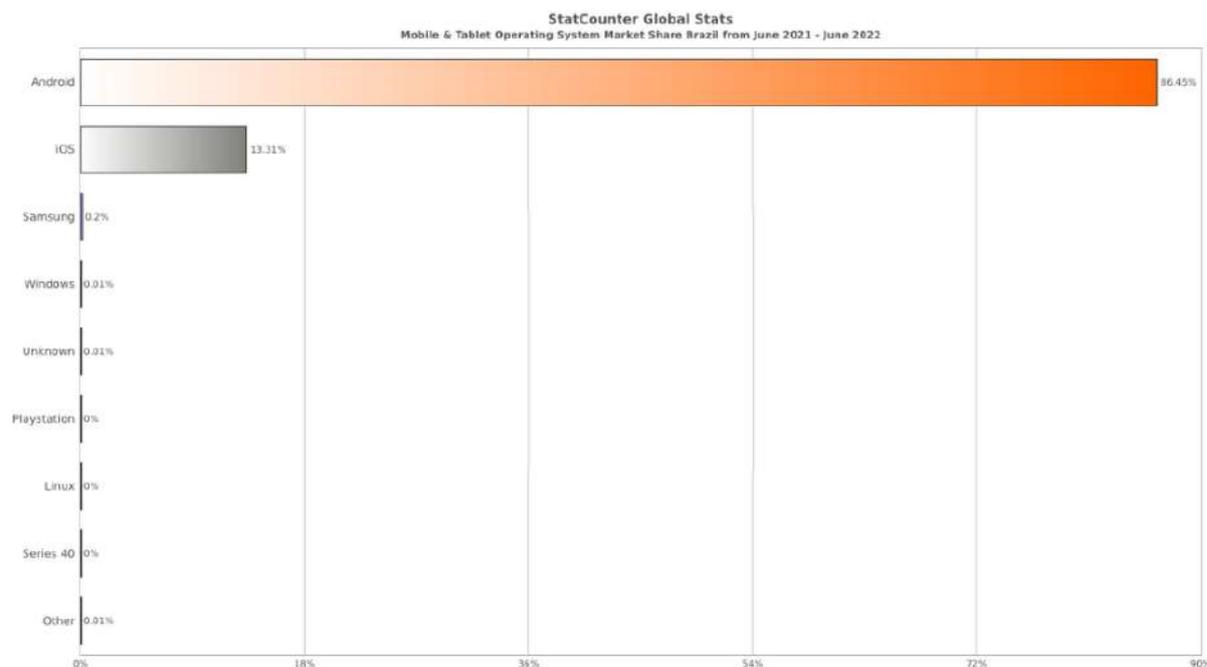


Figura 8 – Participação de mercado de sistemas operacionais para celulares e tablets no Brasil. Julho 2021 a Julho 2022

Fonte: <<https://gs.statcounter.com>>

Além disso, existe o chamado SDK (Software Development Kit) do Android e a IDE (Integrated Development Environment) Android Studio que fornece as ferramentas e ambientes necessários para construir aplicativos Android. Estes recursos e mais informações estão disponíveis gratuitamente no site Android Developers.

2.3.2.1.1 Java

Java é uma linguagem de programação orientada a objetos e possui uma enorme quantidade de bibliotecas de classe que ajudam a desenvolver aplicativos poderosos de forma ágil. Os aplicativos Android são desenvolvidos principalmente com a linguagem de programação Java e sua escolha foi uma escolha lógica devido ao seu poder, gratuita, de código-fonte aberto e milhões de desenvolvedores pelo mundo (DEITEL; DEITEL; DEITEL, 2015).

Segundo Furgeri (2010) a linguagem Java possui diversas características que podem gerar páginas e páginas de conceitos, mas destaca em sua obra as principais características que são: orientação a objetos, portabilidade e suporte à comunicação em rede.

- Orientação a objetos: é um paradigma de programação que traz a ideia de objetos como sendo a representação do mundo real, como um automóvel, uma casa, uma pessoa etc. Java apresenta o conceito de classe, definida por um trecho de código a partir do qual os objetos são criados, dentre várias outras características.

- Portabilidade: por possuir uma máquina virtual Java, torna-se uma linguagem multiplataforma, portanto uma mesma aplicação pode ser executada em diferentes tipos de plataforma sem a necessidade de adaptação de código permitindo que um programa escrito em Java seja executado em qualquer sistema operacional que possua uma máquina virtual instalada.
- Suporte à comunicação: uma das vantagens do Java é fornecer um grande conjunto de classes com funcionalidades para programação em rede, o que agiliza a implementação tecnologias avançadas de comunicação, como protocolos TCP/IP (Transport Control Protocol/Internet Protocol), HTTP, FTP (File Transfer Protocol), entre outros.

2.4 A Técnica Leito Fluidizado (Air Roasting)

A técnica leito fluidizado, também conhecida como Air Roasting, no contexto do café, consiste em expor os grãos de café ao calor e mantê-los em flutuação para que o calor seja passado para os grãos na forma de convecção (SCHENKER; STEFAN, 2000).

Segundo KESTERING (2016), a técnica leito fluidizado (agitado) é um leito de partículas através do qual escoam um fluido, mantendo-as em suspensão, ou seja, no estado de fluidização. O escoamento gás-sólido em leito fluidizado pode ocorrer em várias operações industriais como secagem, craqueamento catalítico, combustão, gaseificação e outros.

Para Fox et al. (2014), algumas das vantagens da utilização deste sistema são: uma melhor mistura do fluido, altos coeficientes de transferência de calor e massa e a disponibilização da área superficial das partículas para a transferência, assim obtendo melhores resultados.

Torradores de leito fluidizados existem desde o início dos anos 1970 quando o engenheiro químico Michael Sivetz percebeu que poderia adaptar um processo usado para secar pellets de magnésio para torrar café, inventando assim a torrefação em leito fluidizado. Esse método durante anos ficou conhecido como “torradores Sivetz” (MOTT, 2021).

À medida que o ar quente flui através do torrador, a força levanta os grãos no ar, fazendo com que flutuem. Isso significa efetivamente que os grãos de café estão descansando em uma “cama” de ar aquecido, que é de onde a torrefação em leito fluido recebe o nome (MOTT, 2021).

3 Desenvolvimento

O desenvolvimento consistiu na utilização de uma pipoqueira elétrica, uma vez que ela tem as características de ventilação e aquecimento necessárias para tornar a torrefação por tipo leito fluidizado possível, pois os grãos de café obterão o calor gerado pela resistência da pipoqueira, juntamente com a ventilação que os manterão em movimento e flutuação, conforme explicitado na seção 2.4.

Para que a torra fosse realizada de forma controlada, ou seja, seguindo os valores de temperatura definidos pelo usuário, serão adaptados os circuitos da pipoqueira elétrica e então realizado o controle dos processos de ventilação e aquecimento de forma separadas. Esse modelo de controle pode ser representado como um sistema realimentado, conforme descrito na seção 2.2. Para isso, serão utilizados:

- Um microcontrolador: implementa o controle PID e é a ponte com o mundo externo (nuvem), uma vez que seus módulos oferecem conectividade Wi-Fi. Isso torna possível o uso do protocolo HTTP para enviar e receber requisições pelo padrão *request/response* do protocolo.
- Um sensor de temperatura, que realiza a realimentação do sistema de controle ao medir a temperatura de saída do processo.
- Os relés eletromecânico e de estado sólido, que são os atuadores do sistema de controle, atuando diretamente nos circuitos de ventilação e aquecimento, respectivamente.

A Figura 9 demonstra como os componentes estão representados na malha de controle realimentada, conforme mostrado em teoria pela Figura 6. O processo da figura abaixo diz respeito ao circuito de aquecimento da pipoqueira, no qual a resistência interna é a representação do processo.

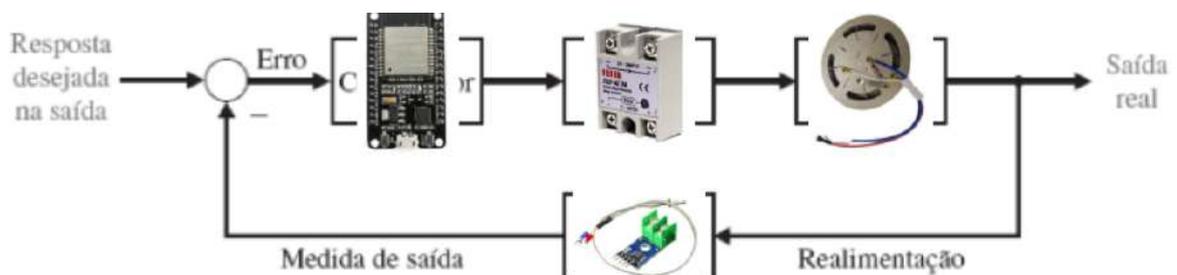


Figura 9 – Componentes do projeto na malha de controle - Circuito de Aquecimento
Fonte: Autoria Própria

Além disso, o microcontrolador recebe os dados do sensor de temperatura e os envia para um banco de dados de tempo real armazenado em nuvem (Firebase), que permite o

acompanhamento da torra em tempo real por um aplicativo mobile. O aplicativo também permite ao usuário definir as curvas de torra, acessar torras públicas de outros usuários, dentre outras funcionalidades que serão mostradas adiante.

A Figura 10 apresenta a arquitetura do projeto. O desenvolvimento do projeto consistiu em duas partes fundamentais: desenvolvimento de hardware e de software. A primeira parte visa detalhar os componentes utilizados, os circuitos elétricos e a montagem do protótipo. A segunda consiste na programação do microcontrolador e desenvolvimento mobile, bem como suas conexões com o Firebase.



Figura 10 – Arquitetura do projeto
Fonte: Autoria Própria

3.1 Desenvolvimento do Hardware

Nesta seção, serão abordados assuntos referentes aos materiais utilizados para confecção do protótipo, como a pipoqueira elétrica, microcontrolador, sensores, relé de estado sólido (*Solid State Relay - SSR*) e demais componentes eletrônicos utilizados no desenvolvimento. Também será apresentado um esquemático completo do circuito, utilizando os componentes citados e por fim a montagem do protótipo. A montagem do protótipo foi adaptado de Costa (2018), porém utilizando-se do conceito de Internet das Coisas uma vez que neste projeto o microcontrolador tem módulo de conectividade Wi-Fi. Também foi implementado o controle PID dentro do próprio código do microcontrolador. Além disso, foi desenvolvido um serviço com aplicativo móvel para definição de curvas de torra e acompanhamento do processo em tempo real e demais funcionalidades que serão mostradas na seção 3.2.

3.1.1 Materiais utilizados

3.1.1.1 Pipoqueira Elétrica

Como dito anteriormente na seção 2.4, para que exista o processo de torrefação adequada, utilizando a técnica de leito fluidizado, foi utilizado uma pipoqueira elétrica da marca Britânia, modelo 62551001, voltagem 127V, potência 1100 W, consumo 1,1 KW/h e peso 980g.



Figura 11 – Pipoqueira Britânia Pop Time
Fonte: <https://britania.com.br>

3.1.1.2 Microcontrolador

Com a popularização dos módulos da marca Espressif, a família dos microcontroladores ESP8266 começaram a surgir no mercado a partir de 2014, onde sua principal característica é oferecer microcontroladores com Wi-Fi embutido, de baixo custo. Após os ESP8266, vieram os mais poderosos microcontroladores ESP32 que possuem diversas vantagens em relação ao seu antecessor, como por exemplo processador dual core, mais pinos, mais memória, Wi-Fi, *Bluetooth*, e etc. Para a realização do projeto, foi utilizada a placa NodeMCU ESP-WROOM-32. Na Tabela 2 as são descritas as especificações técnicas do modelo citado.

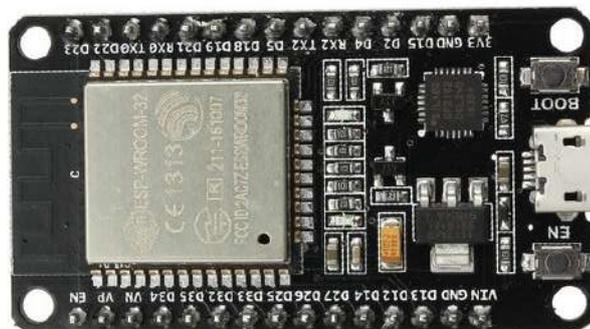


Figura 12 – NodeMCU ESP-WROOM-32
Fonte: [Usinainfo](https://www.usinainfo.com)

Tabela 2 – Características do microcontrolador NodeMCU ESP-WROOM-32

Modelo NodeMCU	NodeMCU ESP-WROOM-32
Tamanho NodeMCU	50mm x 36mm
Memória Flash/SRAM	16 MB / 512 KB
Pinos de E/S digitais	36
Pinos de entrada analógica	18
Wi-Fi integrado	802,11 b/g/n
Bluetooth integrado	v4.2 BR / EDR e BLE (<i>Bluetooth Low Energy</i>)
Faixa de temperatura	-40 °C – 125 °C
Tensão de operação	2.2V – 3.6 DC (5V via Micro USB)

3.1.1.3 Sensor de Temperatura

Para a realização do protótipo, foi necessário um sensor de calor que suporte altas temperaturas. Para isso, foi utilizado o termopar tipo K com o módulo MAX6675 (Figura 13), que realiza medições entre 0°C e 600°C com precisão de 0,25°C e erro máximo de ± 5 °C. Sua temperatura de operação está entre -20 °C e 85°C.



Figura 13 – Max6675 e termopar tipo K
Fonte: <https://www.filipeflop.com>

3.1.1.4 Relés Eletromecânico e de Estado Sólido

Relés são as chaves de comando que operam com sinais elétricos de baixa potência e controlam a maioria dos tipos de circuitos. Analogamente, é como pensar num interruptor de parede, porém para os circuitos eletrônicos. Existem alguns tipos de relés mais conhecidos, são os relés eletromecânicos e de estado sólido. Os relés mais comuns são os eletromecânicos (Figura 14), que é formado por um eletroímã e ao receber um estímulo elétrico, movimenta uma parte mecânica que a posição do contato de uma posição para outra. Este foi utilizado para o circuito de ventilação do protótipo.

O relé de estado sólido tem funcionamento diferente, nele o eletroímã é substituído por um emissor de luz e os contatos são substituídos por dispositivos sensíveis à luz. Quando é recebido o estímulo elétrico, o receptor sensível a luz detecta o sinal luminoso do emissor

e então comuta um dispositivo de maior potência, como TRIACs ou MOSFET de potência (HELDT, 2019). A vantagem do SSR ao eletromecânicos é não possuir componentes mecânicos, conseqüentemente sua comutação é mais rápida e não há desgaste por atrito, resultando em uma maior vida útil. A desvantagem é o calor gerado pelo SSR, uma vez que o dispositivo de maior potência emite muita energia em forma de calor.

Para o projeto, o dispositivo responsável pela atuação da resistência da pipoqueira foi o relé de estado sólido, SSR 40 da marca FOTEK, ideal para aplicações de alta potência como o acionamento de resistências aquecedoras da pipoqueira. Sua tensão de operação é de 3V até 32V em corrente contínua e a tensão de saída é de 24V até 380V em corrente alternada. Juntamente com o SSR, foi utilizado um dissipador térmico para o modelo da Figura 15.



Figura 14 – Relé eletromecânico
Tongling

Fonte: <https://www.huinfinito.com.br>



Figura 15 – Solid State Relay - SSR
Fotek

Fonte: <https://www.casadarobotica.com>

3.1.1.5 Demais materiais necessários

Além dos principais componentes citados anteriormente, é importante para a prototipação a seguinte lista de materiais.

- *Protoboard*
- Cabos jumper e fios para circuitos
- Solda para circuitos
- Fita isolante
- Furadeira e alicate
- Fonte de 5V para o ESP32 e 24V para o motor DC
- Plug P4 compatível com a fonte 24V

3.1.2 Esquemático do Circuito Elétrico

Neste tópico será abordado a integração dos materiais utilizados, uma vez que já foram apresentados no tópico (3.1.1), em um único circuito, ou seja, o circuito completo da torradeira (Figura 18). Inicialmente, é mostrado o circuito da pipoqueira elétrica (Figura 16), que em seguida foi separado em dois circuitos distintos: o de ventilação e o de aquecimento (Figura 17).

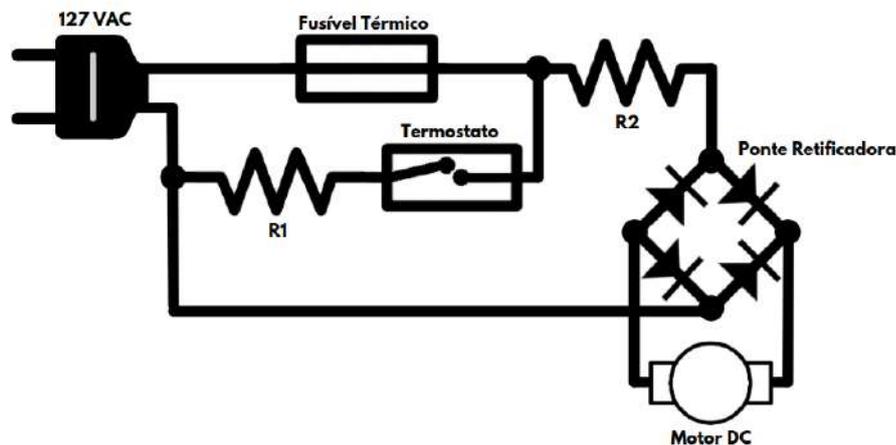


Figura 16 – Circuito Elétrico - Pipoqueira Britânia Pop Time
Fonte: (COSTA, 2018) - Adaptado

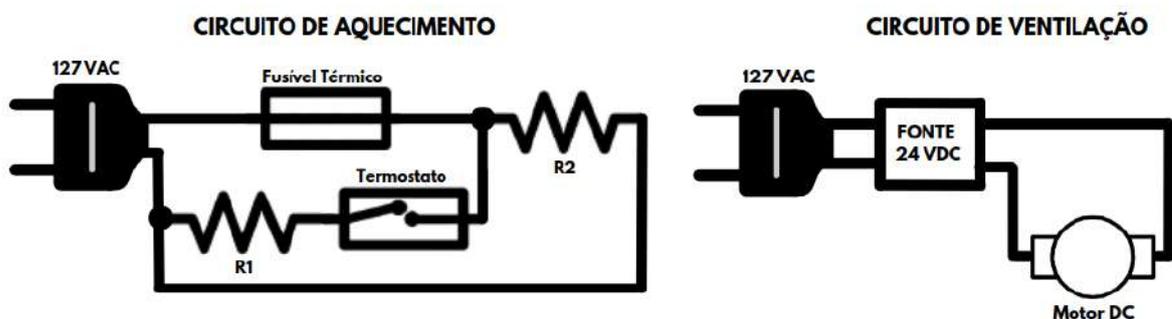


Figura 17 – Circuito Elétrico (Ventilação e Aquecimento Separados) - Pipoqueira Britânia Pop Time

Fonte: (COSTA, 2018) - Adaptado

Feito a separação no circuito da pipoqueira, como mostrado na Figura 17, pode-se adicionar o microcontrolador, o SSR, o termopar e o relé eletromecânico. O controle do aquecimento será feito através do SSR, fazendo com que o relé ligue e desligue conforme o ajuste do controlador implementado pelo microcontrolador. Da mesma forma, para o circuito de ventilação será utilizado o relé eletromecânico para atuar no processo. O esquema completo está ilustrado na Figura 18.

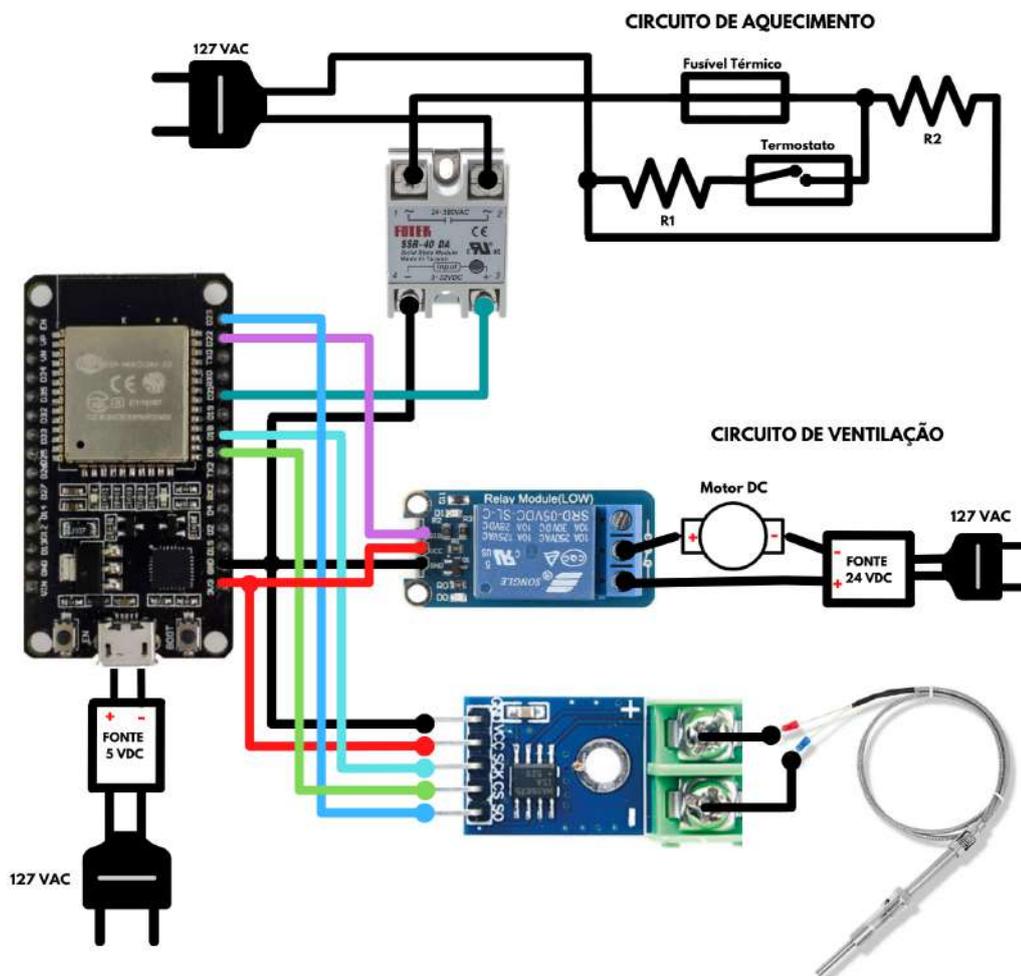


Figura 18 – Circuito Elétrico Completo
Fonte: Autoria própria

3.1.3 Montagem do Circuito Completo

Primeiramente foi necessário abrir a pipoqueira elétrica, desparafusando os 4 parafusos que estão na parte de baixo da pipoqueira. Após abri-la, foi identificado o motor DC e os quatro diodos. A partir daí, bastou realizar a separação dos circuitos, conforme mostrado no esquemático das Figuras 16 e 17.

Nas Figuras 19 e 20 é mostrada a separação dos circuitos. Note que os fios brancos da Figura 19 e também que os diodos do circuito original ficaram em desuso. Isso porque esses diodos formam um “Diode bridge” que transformam a corrente alternada em corrente contínua, entretanto como a alimentação foi utilizada com a fonte de corrente contínua, dispensou o uso desses diodos, bem como os fios brancos e este bastou isolá-los.

A Figura 21 mostra como foi feita a adaptação de dois fios (azuis) que serão alimentados pela fonte de 24V DC ligadas ao relé eletromecânico. Neste passo, foi realizada a soldagem dos fios nos terminais do motor DC. Vale destacar a tensão positiva e negativa



Figura 19 – Montagem original
Fonte: Próprio Autor



Figura 20 – Separando os circuitos
Fonte: Próprio Autor

do motor DC, em que a positiva é notada por um pequeno círculo ao lado da ligação do fio azul à direita.



Figura 21 – Ligação dos fios para o motor DC
Fonte: Próprio Autor

Ainda com a pipoqueira aberta, foi realizado um furo, com o auxílio de furadeira utilizando uma broca de 5 milímetros, para acoplar o sensor de temperatura em sua carcaça com finalidade de obter uma melhor precisão de medida de temperatura, conforme Figura 22. Vale destacar, que como o sensor vem com uma barra roscada, bastou rosquear um pouco para que fique bem fixa na carcaça. Além disso, o botão tic-tac da pipoqueira (encontra-se na parte lateral) foi retirado para que seja possível passar os fios azuis do motor DC e o fio do sensor de temperatura. Por fim, os fios conectados no botão tic-tac

serão usados para conexão ao SSR.

Em seguida, foi fechada a pipoqueira conforme mostrado na Figura 23. Note que os fios citados acima foram passados pelo buraco do botão tic-tac e os fios do botão tic-tac foram realizadas emendas para aumentar o tamanho dos fios.



Figura 22 – Furo de acoplamento do sensor de Temperatura
Fonte: Próprio Autor



Figura 23 – Passagem dos fios por debaixo da pipoqueira
Fonte: Próprio Autor

Após esse momento, foi montado o circuito completo conforme o esquemático da Figura 18. A Figura 25 ilustra esses procedimentos e coloca os equipamentos mostrados na Figura 24 dentro de uma caixa para melhorar a estética do protótipo.

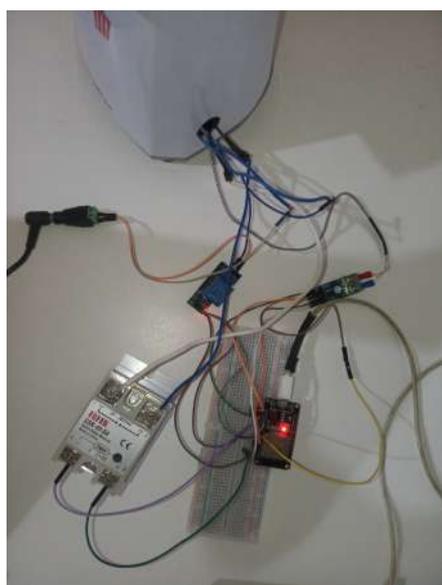


Figura 24 – Circuito completo - montagem
Fonte: Próprio Autor



Figura 25 – Estética final da montagem
Fonte: Próprio Autor

Desta forma, a primeira parte do projeto (desenvolvimento de Hardware) foi realizada.

3.2 Desenvolvimento do Software

Nesta seção serão abordadas as ideias gerais referentes a elaboração do programa do microcontrolador, bem como as telas e funcionalidades do aplicativo mobile.

Para o desenvolvimento do software, inicialmente foi criado um projeto denominado “torrador-app” no site do [Firebase](#). Em seguida, foram ativados os seguintes produtos disponíveis para o projeto: Realtime Database, que é o banco de dados de tempo real e o Authentication, que permite o aplicativo reconhecer a identidade do usuário de forma simples e segura.

Vale destacar que todos os serviços/produtos do Firebase utilizados compreendem o plano gratuito (Tabela 1), visto que atende aos propósitos deste projeto.

3.2.1 Programação do Microcontrolador

A etapa de programação do microcontrolador foi realizada através do software de código aberto Arduino IDE, disponível para os sistemas operacionais Mac, Windows e Linux. Essa IDE, embora inicialmente usada para arduinos, também oferece suporte para as placas NodeMCU ESP-WROOM-32, através de seu Gerenciador de Placas.

Além disso, é possível adicionar bibliotecas através do seu Gerenciador de Bibliotecas, justamente por que as bibliotecas são muito úteis para adicionar as funcionalidades dos módulos do microcontrolador, como por exemplo a biblioteca Wi-Fi. Desta forma, o uso desse ambiente de desenvolvimento integrado facilita a escrita de código, em linguagem C, e o upload para a placa.

Inicialmente, para a primeira etapa no desenvolvimento programação do microcontrolador, foi necessária a instalação da IDE do arduino e a configuração do microcontrolador ESP32.

Para o projeto, foi necessário a instalação das bibliotecas:

- [ESPAsync_WiFiManager](#): biblioteca responsável por lidar com as configurações das credenciais WiFi dos módulos ESP32, ESP8266 de maneira assíncrona. Devido a essa biblioteca, torna-se desnecessário a configuração das SSID (nome da rede) e PASSWORD (senha da rede) do WiFi no código do programa.
- [max6675](#): biblioteca que implementa a usabilidade do sensor de temperatura.
- [PID](#): biblioteca que implementa o controlador clássico discreto.

- *AsyncElegantOTA*: biblioteca que fornece uma interface WEB para fazer upload de atualizações de arquivos '.bin' via OTA (*Over the Air*) para módulos ESP8266 e ESP32, com status preciso e progresso exibidos na interface do usuário.
- *Firebase_ESP_Client*: biblioteca que permite a conexão assíncrona com o banco de dados *RealTime-Database* Firebase do Google.

No código do microcontrolador foram feitas as definições das bibliotecas mencionadas acima, das variáveis globais, úteis para a lógica de programação e das variáveis de acesso do Firebase.

Na função de configuração, denominada como função *setup*, é feita a conexão WiFi, em que é obtido um IP para o microcontrolador. Também é feita a inicialização do servidor Web que permite invocar a biblioteca de atualização via OTA e a criação de uma página de “confirmação” para permitir consultar se o dispositivo é uma torradeira. Ainda na função *setup* é realizada a conexão com o banco de dados Firebase, onde é definido a função de *callback* do banco de dados. Por fim, são inicializados os parâmetros da biblioteca PID que definem o controle clássico discreto a fim de manter a temperatura do sensor ajustada de acordo com o desejado.

Na função *loop*, são realizadas as lógicas de programação para o correto funcionamento do programa. Vale destacar que o microcontrolador, ao conectar-se ao Firebase, envia seus dados de configuração para o banco de dados e então é criada uma nova estrutura que contém as informações referentes ao microcontrolador, explicitado em 3.2.3

Outra importante função no código é a de *callback* do Firebase. Essa função detecta mudanças nos nós do banco de dados, ou seja, a função fica “ouvindo” o banco de dados. Desta forma, ao perceber alguma mudança, pode ser realizado as lógicas de programação para que o microcontrolador atue, como por exemplo, ligar os circuitos de ventilação e aquecimento.

Todo o código do microcontrolador está disponível no [repositório online](#).

3.2.1.1 Configuração do Firebase no Microcontrolador

Para a configuração do banco de dados Firebase no código do microcontrolador, foram feitas as definições das variáveis de acesso, nas linhas 46, 47, 48 e 49.

```
#define API_KEY ""  
#define DATABASE_URL ""  
#define USER_EMAIL ""  
#define USER_PASSWORD ""
```

No projeto “torrador-app” criado na seção 3.2, foi obtida a chave de API Web (Figura 26), que foi utilizada no código do microcontrolador na linha 46 para definir a

APY_KEY. De modo semelhante, a DATABASE_URL foi obtida conforme a Figura 27 e por fim, o USER_EMAIL e USER_PASSWORD, conforme a Figura 28. Com essas variáveis configuradas, o microcontrolador cria a conexão com o banco de dados do projeto.



Figura 26 – Obtendo API_KEY do Firebase
 Fonte: Autoria própria

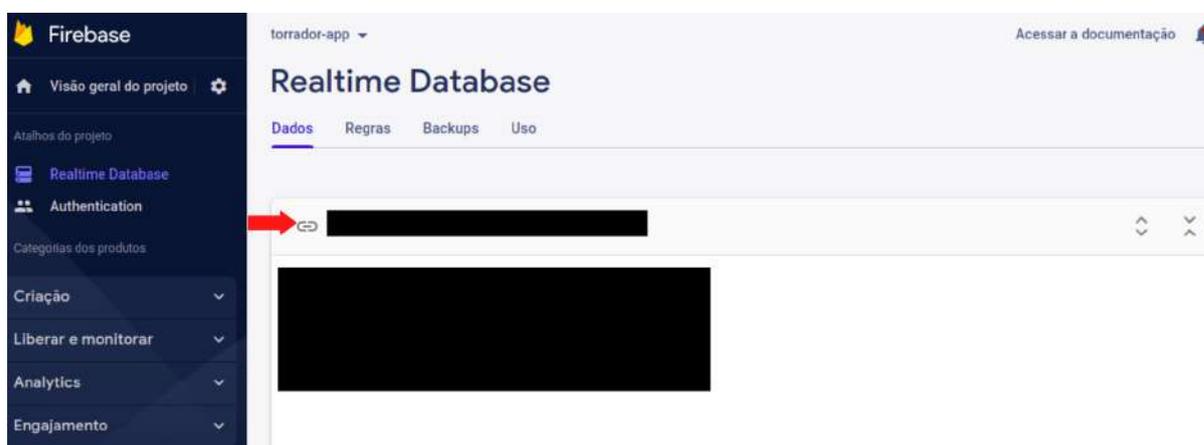


Figura 27 – Obtendo DATABASE_URL do Firebase
 Fonte: Autoria própria

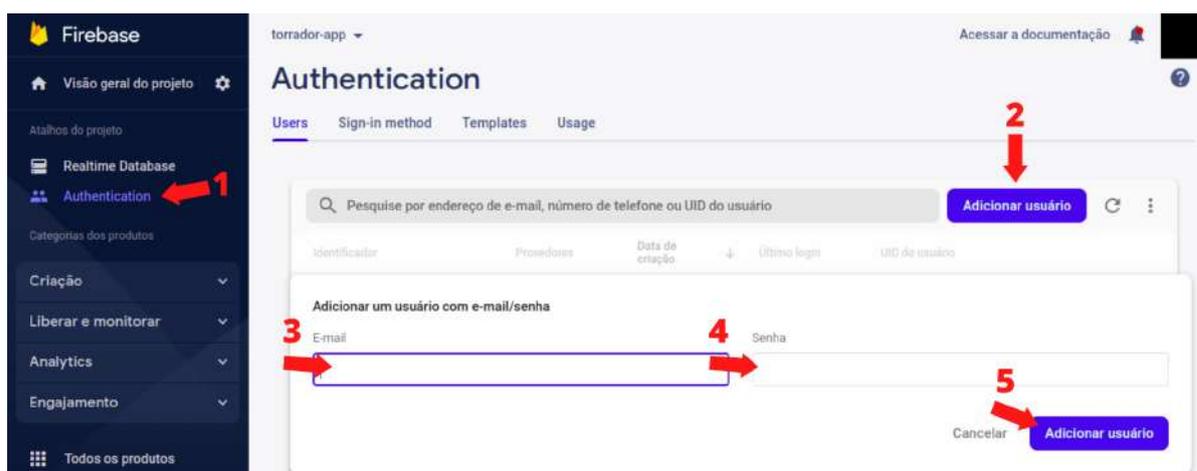


Figura 28 – Obtendo USER_EMAIL e USER_PASSWORD do Firebase
 Fonte: Autoria própria

3.2.2 Desenvolvimento Mobile

A etapa de desenvolvimento mobile para o celulares Android foi realizada através do software IDE Android Studio. O código está disponível no [repositório online](#).

Inicialmente, foram feitas as configurações do aplicativo com o Firebase. Para isso, bastou abrir o projeto pelo Android Studio e clicar na barra superior em Tools -> Firebase -> RealTime Database.

Após isso, foi feita a conexão com o Firebase, selecionando o projeto “torrador-app” e na sequência foi adicionado o Realtime Database SDK no aplicativo, conforme mostrado na Figura 29. Com isso feito, o aplicativo estava apto a realizar as operações de leitura e escrita no banco de dados.

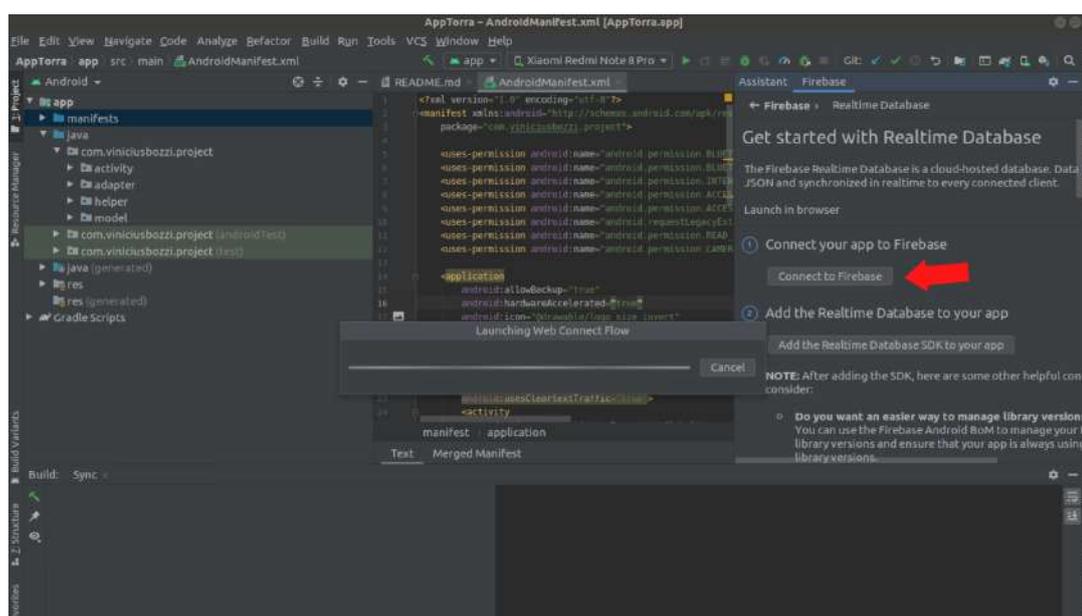


Figura 29 – Configuração de conexão do Aplicativo e Firebase
 Fonte: Autoria própria

O aplicativo permite as seguintes funcionalidades: login, acessar torras particulares, cadastrar torradeiras, criar torras, acessar de torras compartilhadas e acompanhar torra em tempo real. A Figura 30 apresenta o diagrama de casos de uso.

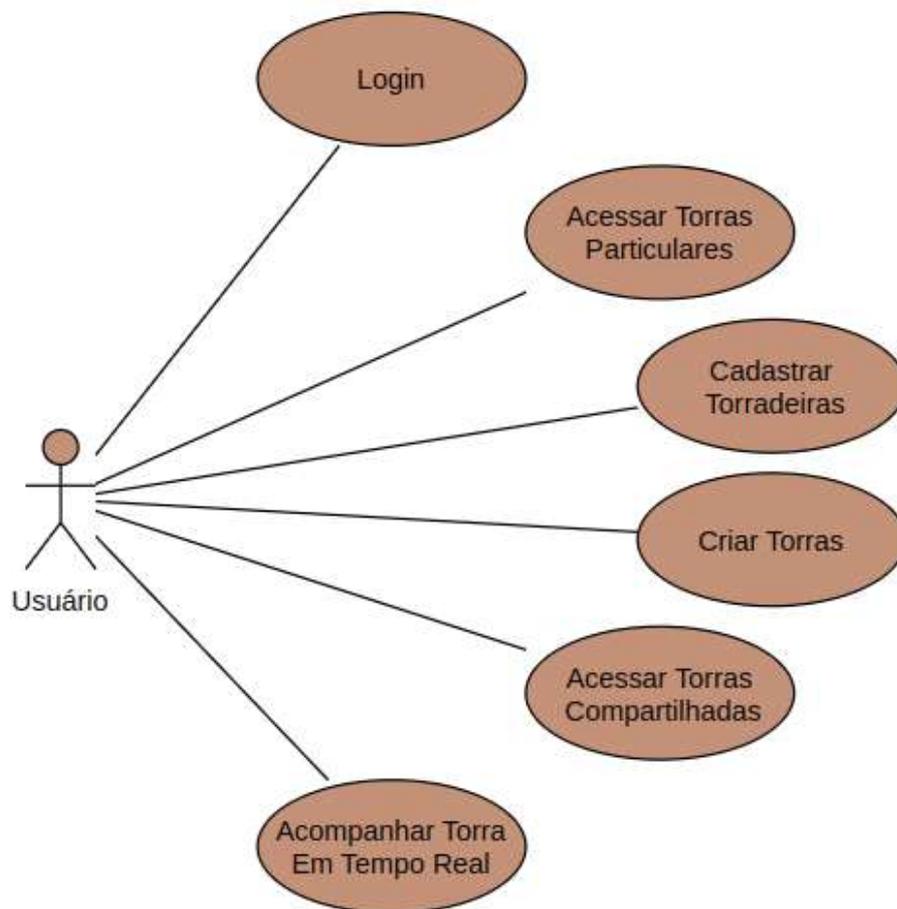


Figura 30 – Diagrama de casos de uso
Fonte: Autoria própria

Serão mostradas nas subseções adiante as telas do aplicativo que contém as funções apresentadas pelo diagrama de casos de uso.

3.2.2.1 Login

Ao abrir o aplicativo, há uma tela splash, representada pela Figura 31, que carrega processos de conexão em background, como por exemplo a autenticação do usuário. Caso a autenticação de usuário já tenha sido feita, o aplicativo entra diretamente na tela principal “Minhas Torras”, representada na Figura 33, ou então entra em uma tela de login ou cadastro para que o usuário entre na sua conta ou cadastre-se, respectivamente. (Figura 32).



Figura 31 – Tela Inicial Splash
Fonte: Próprio Autor



Figura 32 – Tela de Login e Cadastro
Fonte: Próprio Autor

3.2.2.2 Acessar Torras Particulares

Na tela principal (Figura 33) é mostrada a lista de torras particulares do usuário. Nessa tela existem as opções de acessar cada torra, excluí-las se o toque na tela persistir por 2 segundos e adicionar uma nova torra (Subseção 3.2.2.3) ao clicar no botão no canto inferior direito da tela. Também nesta tela (Figura 34) existem três pontos no canto superior direito, com as opções de Torras Globais (Subseção 3.2.2.5), que mostra as torras públicas de todos os usuários; Dispositivos (Subseção 3.2.2.4), que cadastra uma torradeira para o usuário enviar as curvas de torra para torrar; e Sair, que desloga o usuário.



Figura 33 – Tela Principal (a)
“Minhas Torras”
Fonte: Próprio Autor

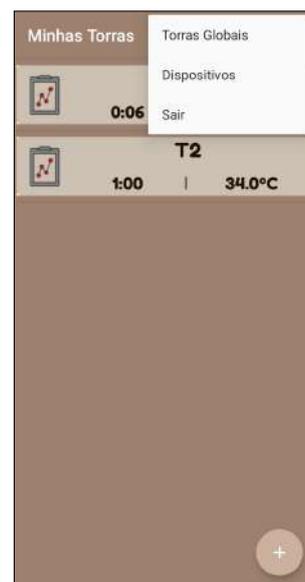


Figura 34 – Tela Principal (b)
“Minhas Torras”
Fonte: Próprio Autor

3.2.2.3 Criar Torras

A Figura 35 corresponde a tela de cadastro de novas torras. Para isso, deve-se definir um nome para a torra, declarar se ela é privada ou pública e adicionar os pontos no gráfico de temperatura (em graus Celsius) pelo tempo (em minutos). Para a definição dos pontos no gráfico, é permitido temperaturas de 290 graus Celsius e tempo máximo 15 minutos, visto que a pipoqueira permite até 17,5 minutos de aquecimento sem precisar esfriar. Após realizar a definição destes pontos citados, é necessário clicar no botão superior direito para salvar a nova curva de torra.



Figura 35 – Tela de Adicionar Nova Torra
Fonte: Próprio Autor

3.2.2.4 Cadastrar Torradeiras

Na tela de configurações dos Dispositivos existem duas formas de cadastro de torradeiras.

A primeira (Figura 36) é a listagem de torradeiras na rede. Neste caso o smartphone deve estar na mesma rede para que seja enviado ping's testando a conectividade entre os equipamentos e abrindo requisições HTTP com os endereços IP's encontrados.

Uma vez que a torradeira esteja disponível na rede, ela abre um Web Server que cria uma página Web para que o aplicativo possa consultá-la, tornando possível a sua identificação e listagem na tela.

A segunda opção de cadastro (Figura 37) é a leitura via QR Code disponível na carcaça da torradeira. Para este caso não é necessário a torradeira e o smartphone estarem na mesma rede. A identificação única de cada torradeira é dada pelo número MAC de cada microcontrolador ESP32.

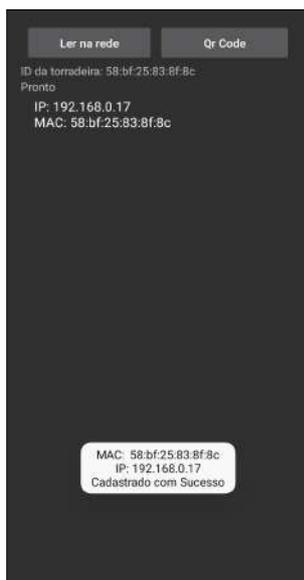


Figura 36 – Tela de Cadastro de Dispositivos - Ler na rede
Fonte: Próprio Autor



Figura 37 – Tela de Cadastro de Dispositivos - QR Code
Fonte: Próprio Autor

3.2.2.5 Acessar Torras Compartilhadas

A Figura 38, denominada como Torras Globais, permite ao usuário o acesso de todas as torras de outros usuários, no caso em que as mesmas estejam marcadas com a opção de torras públicas, conforme visto na subseção 3.2.2.3. Nesta tela é permitido acessar as torras de outros usuário para realizar a torras e exportar os dados da torra. Entretanto, não é possível excluí-las nem editá-las.

Torras Globais	
	TESTEM 2:00 120.0°C
	TORRA 1 6:00 100.0°C
	TORRA 3 12:00 165.0°C
	T2 2:00 55.0°C
	T3 5:00 85.0°C

Figura 38 – Tela Torras Globais
Fonte: Próprio Autor

3.2.2.6 Acompanhar Torra em Tempo Real

A Figura 39 mostra a tela da torra selecionada, isto é, corresponde a torra que o usuário clica nas telas de torras particulares ou de torras compartilhadas. Nela encontram-se as opções de editar a curva de torra pressionando o botão do lápis, centralizado entre

os botões na tela. Pode-se também exportar um arquivo no formato ‘.csv’ que representa a curva de torra (Graus Celsius por Segundo), clicando no botão mais à direita da tela. Por fim, pode-se enviar a curva de torra para a torradeira ao pressionar no botão “G” (Figura 40).



Figura 39 – Tela de Torra Seleccionada
- Visão Geral
Fonte: Próprio Autor



Figura 40 – Tela de Torra Seleccionada
- Enviar curva de torra
Fonte: Próprio Autor

Desta forma, é feito o envio dos dados da curva de torra ao banco de dados para atualizar as informações da torra que posteriormente será identificado pela função de *callback* da torradeira. Após esse envio de dados, o botão "PRÉ-AQUECER" torna-se disponível (Figura 40). Ao pressionar o botão de pré-aquecimento a torradeira toma a ação de ligar os circuitos de ventilação e aquecimento. Essa ação permite que a torradeira seja pré-aquecida até a temperatura desejada e a mensagem "AJUSTANDO TEMPERATURA" fica disponível na tela, conforme Figura 41. Também neste instante a temperatura em tempo real do sensor presente na torradeira fica disponível em tela (no canto superior direito da tela).

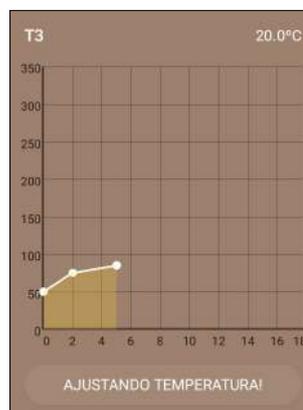


Figura 41 – Tela de Torra Seleccionada - Pré-aquecendo
Fonte: Próprio Autor

Após atingir a temperatura ideal no pré-aquecimento, torna-se disponível o botão

de "INICIAR TORRA", conforme mostrado na Figura 42. Neste momento, deve-se colocar os grãos de café dentro da torradeira e clicar no botão. Por fim, com a torra em andamento, é possível acompanhá-la em tempo real, conforme ilustra a Figura 43. Com o término da torrefação, pode-se compartilhar em formato 'csv' os dados da torra, bem como finalizar. Vale ressaltar que caso o usuário saia da torra antes da sua conclusão ou no caso em que a torra finalize, o circuito de ventilação da torradeira continuará ligado por 100 segundos para o resfriamento, por motivo de segurança.



Figura 42 – Tela de Torra Selecionada - Iniciar Torra

Fonte: Próprio Autor

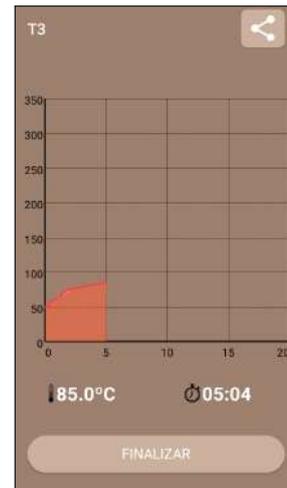


Figura 43 – Tela de Torra Selecionada - Fim da Torra

Fonte: Próprio Autor

3.2.3 Estrutura do Banco de Dados

A estruturação do banco de dados NoSql foi realizada conforme segue a árvore JSON conforme mostrada abaixo. Essa estrutura dos dados foi adotada de forma a lidar com os multiusuários do aplicativo e também conter as informações das torradeiras.

```
{
  "Usuário1": {
    "Torra1": {
      "global": true,
      "idFirebase": "-N4xScQMGPqMuCS0sjZI",
      "nomeGrafico": "testem",
      "temperatura": 120,
      "tempo": 2,
      "valorXY": "50.0,0.0,120.0,120.0"
    },
    "Torra2": { ... },
    ...
    "TorraN": { ... },
    "u_dispositivos": "ID-MAC-Torradeira"
  },
  "Usuário2": {
    "Torra1": { ... },

```

```

    "Torra2": { ... },
    ...
    "TorraN": { ... },
    "u_dispositivos": "ID-MAC-Torradeira"
  },
  ...
  "UsuárioN": { ...
  },
  "dispositivos": {
    "ID-MAC-Torradeira1": {
      "aquecendo": false,
      "dados": "50.0,0.0,200.0,120.0",
      "ip": "10.0.0.105",
      "resfriando": false,
      "temperatura": 20,
      "torrando": false
    },
    "ID-MAC-Torradeira2": { ... },
    ...
    "ID-MAC-TorradeiraN": { ... }
  }
}
}

```

A Figura 44 exemplifica a estrutura do banco de dados. Para estruturar os multiusuários é criado um novo nó para cada novo usuário, em outras palavras, os números 1 da Figura 44 representam cada usuário cadastrado.

De forma análoga, dentro de cada usuário existem suas torras, representado pelo número 2. Além disso, cada usuário pode ter um dispositivo cadastrado, que neste caso é a torradeira, representado pelo número 3. Todas essas operações de escrita no banco de dados são realizadas pelo aplicativo.

Para estruturar as torradeiras, existe um nó denominado “dispositivos” que armazena todas as torradeiras cadastradas, que são identificadas pelo endereço MAC do microcontrolador, conforme representado pelo número 4 da Figura 44. Essas torradeiras são escritas no banco de dados pelo próprio código do microcontrolador.

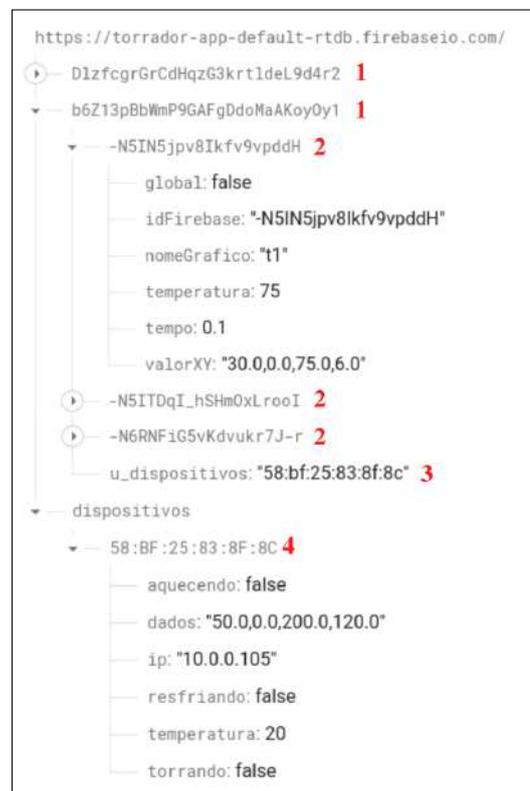


Figura 44 – Estrutura do Banco de Dados

Fonte: Autoria própria

4 Validação da Proposta

Para realizar a avaliação e desempenho do protótipo, inicialmente foram feitas as configurações no aplicativo para a criar um novo usuário, adicionar suas torras e realizar o cadastro da torradeira. Após isso, foram feitos os testes de torrefação que foram organizados da seguinte maneira:

- Testagens para sintonia do controlador: consiste no ajuste do controlador PID para assim avaliar a resposta à curva de torra definida pelo usuário. Para este ajuste foi adotada uma curva de torra com temperaturas menores para evitar que configurações erradas pudessem levar a instabilidade do controle e eventual dano ao equipamento.
- Testagem de perfil de torra: foi realizada com o controlador PID já ajustado de maneira adequada. Com isso, foi possível definir pelo aplicativo uma curva de torra com temperaturas de até 200 graus Celsius para testagem dos perfis de torra reais.

Além disso, foram realizados levantamentos dos custos diretos relacionados à montagem do protótipo e também feito um comparativo com equipamentos similares encontrados no mercado. Também vale destacar que durante todo o processo de validação do proposta foram adotadas medidas de segurança visto que a temperatura da torradeira pode causar acidentes. Por isso, a nível de hardware, foi estipulado que a torradeira não permita temperaturas acima de 200 graus Celsius. Além disso, todo o processo da torra foi feito de forma acompanhada para detectar eventuais anormalidades e não permitir que o circuito de aquecimento permanecesse ligado sem que o circuito de ventilação estivesse ligado, uma vez que isso danificaria o equipamento em alguns segundos.

4.1 Configurações no Aplicativo

Inicialmente, foi criado um novo usuário na tela de cadastro, com e-mail e senha, conforme mostra a Figura 45. Em seguida, foi criada uma nova torra denominada “torra 1”, definindo sua curva de torra e selecionando-a como torra pública, representado na Figura 46. Essa torra (“torra 1”) foi utilizada para a sintonia do PID, devido suas temperaturas máximas chegarem até 100 graus Celsius. Além disso, foi feita outra torra denominada “torra 2”, que foi utilizada como curva de perfil de torra real.

Após isso, estava configurado as duas torras no aplicativo, conforme mostrado na Figura 47. Porém, antes de entrar em algumas das torras, é necessário que uma torradeira seja cadastrada. Esta etapa foi realizada conforme descrito na subseção 3.2.2.4 utilizando a listagem de torradeiras na rede, uma vez que a torradeira estava na mesma rede do



Figura 45 – Cadastro de novo usuário
 Fonte: Próprio Autor



Figura 46 – Criação de nova torra
 Fonte: Próprio Autor

smartphone. Com essas etapas realizadas, o aplicativo e a torradeira estavam prontos para a realização da torra.



Figura 47 – Tela principal “Minhas Torras”
 Fonte: Próprio Autor

4.2 Testagem da Torrefação

Para que fosse possível realizar a torra, foram separados 30 gramas de café verde, conforme a Figura 48.

Esta quantidade geralmente é próxima a quantidade necessária para o preparo de três xícaras de café (10 gramas cada xícara). Além disso, essa quantidade permite que os grãos de cafés tenham um bom movimento de rotação no interior da torradeira.



Figura 48 – Café verde para a torra
Fonte: Autoria própria

4.2.1 Testagens para Sintonia do Controlador

Para obter uma melhor fidelidade à curva de torra elaborada pelo usuário no aplicativo, foi necessário realizar a sintonia do PID, conforme explicitado em 2.2.3. Para realizar o ajuste de maneira dinâmica, ou seja, sem que fosse preciso definir as constantes no código do microcontrolador para em seguida compilar e executar o programa, foi criada dentro da função *loop()* uma estrutura que permitiu que as definições das constantes do controle PID fossem passadas pela entrada Serial e deste modo o ajuste pôde ser feito de maneira mais rápida.

Os valores das constantes K_p , K_i e K_d são utilizadas para gerar o sinal de controle para atuar na temperatura da pipoqueira. Sinal esse representado pela Equação (2.4). Para então definir as constantes K_p , K_i e K_d respectivamente pela entrada Serial, basta escrever na Serial PX, IX ou DX, onde X é o valor da constante. A Figura 49 mostra a saída do monitor serial e onde foi definido para a constante K_p do controlador o valor de 4. De mesmo modo foram feitas as definições das outras constantes do controlador durante a execução da “torra 1”, citada anteriormente.

```

P4
INPUT: 81.25
OUTPUT: 0.00
Relé baixo
SETPOINT: 69.00
INPUT: 81.25
OUTPUT: 0.00
Relé baixo
Set float...Set float... ok
SETPOINT: 69.00
INPUT: 81.50
OUTPUT: 0.00
Relé baixo
break path. /dispositivos
event path. /58:BF:25:89:8F:BC/temperatura
data type. float
event type. put
81.25000
Received stream payload size: 79 (Max. 206)
SETPOINT: 69.33
INPUT: 81.75
OUTPUT: 0.00
Relé baixo
SETPOINT: 69.33
INPUT: 81.50
OUTPUT: 0.00
Relé baixo
SETPOINT: 69.33
INPUT: 81.75
OUTPUT: 0.00
Relé baixo
SETPOINT: 69.33
INPUT: 80.75
OUTPUT: 0.00
Relé baixo
SETPOINT: 69.67
INPUT: 81.50
OUTPUT: 0.00
Relé baixo

```

Figura 49 – Definição das constantes do PID pela entrada Serial
Fonte: Autoria própria

No primeiro momento, para a torra 1 foram utilizadas as constantes do PID como

sendo $K_p = 5$, $K_i = 5$, $K_d = 5$, definidas no código do microcontrolador. Para o primeiro teste de ajuste, a torra foi realizada do início ao fim com as constantes já definidas. Foi então acessada a torra 1 pelo aplicativo e enviado os dados da curva de torra para a torradeira ao pressionar o botão “G”, conforme representado pela Figura 50. Em seguida, foi clicado no botão de pré-aquecimento e a torra teve seu seu modo de pré-aquecer até 60 graus Celsius. Ao atingir essa temperatura, foi clicado no botão de Iniciar Torra e foi possível acompanhá-la em tempo real, conforme mostrado na Figura 51.

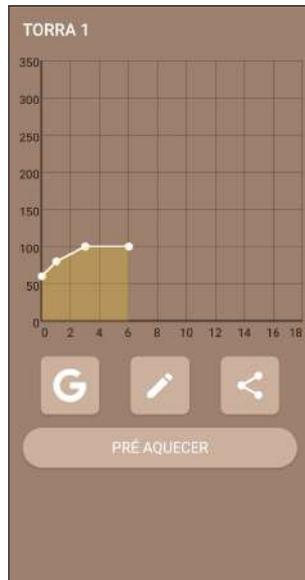


Figura 50 – Tela de Torra Seleccionada “torra 1”
Fonte: Próprio Autor



Figura 51 – Tela de Torra Seleccionada - Fim da Torra - $K_p=5$ $K_i=5$ $K_d=5$
Fonte: Próprio Autor

Ao final da torra, pôde-se exportar os dados reais da torra em cada segundo. Esses dados podem ser acessados pelo repositório [online](#). Pôde-se perceber que os resultados, apesar de acompanhar a curva de torra definida pelo usuário, oscilavam cerca de 10 graus Celsius positivos e 10 graus Celsius negativos em relação ao *setpoint*.

Utilizando-se dos Índices de Desempenho mostrados em 2.2.4, pode-se calcular o ISE e IAE da sintonia do controlador. Vale destacar que quanto mais próximo de zero os valores de IAE e ISE, mais se aproxima a curva de tempo real da curva definida pelo usuário. Inicialmente, deve ser feita a discretização das equações 2.5 e 2.6 uma vez que os valores gerados pela torra são a cada segundo. Então a formulação equivalente discreta é dada pelas equações abaixo:

$$IAE = \int_0^T |E(t)|dt = \sum_{k=0}^N |E[k]| \quad (4.1)$$

$$ISE = \int_0^T E^2(t)dt = \sum_{k=0}^N E^2[k] \quad (4.2)$$

Desta forma foi calculado os Índices de Desempenho para a torra 1 com os parâmetros do PID sendo $K_p = 5$, $K_i = 5$, $K_d = 5$.

$$IAE = \sum_{k=0}^{360} |E[k]| = 2455 \quad (4.3)$$

$$ISE = \sum_{k=0}^{360} E^2[k] = 21374.375 \quad (4.4)$$

Após a torra inicial, foi feita a sintonia do controlador definindo as constantes pela entrada Serial, conforme citado anteriormente. Também foi utilizada a torra 1 como curva de torra de sintonia. Foram repetidos os procedimentos de envio dos dados da curva de torra para a torradeira ao pressionar o botão “G”, foi clicado no botão de pré-aquecimento e após pré-aquecer a torradeira foi iniciada a torra. Neste momento foram feitas as mudanças nas constantes.

Após diversos testes com o mesmo procedimento de definir as constantes pela entrada Serial e utilizando a técnica de controle citado em 2.2.3, foi obtido um resultado mais fiel à curva de torra definida pelo usuário com os valores das constantes do PID de $K_p = 40$, $K_i = 1$ e $K_d = 50$. A Figura 52 mostra os resultados deste ajuste conforme foram acompanhadas em tempo real.



Figura 52 – Tela de Torra Seleccionada - Fim da Torra - $K_p=40$ $K_i=1$ e $K_d=50$

Fonte: Próprio Autor

Pôde-se perceber que a temperatura oscilava cerca de 2 graus Celsius positivos e 2 graus Celsius negativos em relação ao *setpoint*. Os *logs* dessa torra são encontrados no repositório [online](#). Foi também realizado o cálculo dos Índices de Desempenho para a torra 1 com os parâmetros do PID sendo $K_p = 40$, $K_i = 1$, $K_d = 50$.

$$IAE = \sum_{k=0}^{360} |E[k]| = 309.0833333 \quad (4.5)$$

$$ISE = \sum_{k=0}^{360} E^2[k] = 437.3125 \quad (4.6)$$

Desta forma, pôde-se observar a melhora da curva de torra tanto visualmente pelo acompanhamento em tempo real como pelos Índices de Desempenho, visto que a integral dos erros absolutos bem como a integral quadrática dos erros tiveram seus valores reduzidos de 2455 para 309.0833 e 21374.375 para 437.3125, respectivamente.

Com isso, o ajuste do PID com os valores das constantes de $Kp = 40$, $Ki = 1$ e $Kd = 50$ foi utilizado para todas as torras subsequentes deste trabalho, uma vez que obteve um resultado próximo à curva de torra definida pelo usuário.

4.2.2 Testagem de Perfil de Torra

Com o controle ajustado, conforme mostrado na seção anterior, foi possível realizar a testagem de perfil de torra utilizando a “torra 2”. Os mesmos procedimentos no aplicativo foram feitos para enviar os dados da curva de torra e o pré-aquecimento da torradeira. Após pré-aquecer, foi inserido os grãos de café verde (Figura 54) e foi iniciada a torra. A mesma pôde ser acompanhada em tempo real conforme mostrado na Figura 53.



Figura 53 – Acompanhamento em tempo real - “torra 2”

Fonte: Próprio Autor

Pôde-se notar nesta torra que a pipoqueira elétrica apresentou a limitação no circuito de aquecimento. A medição do sensor de temperatura atingiu seu valor máximo de 171 graus Celsius. Por este motivo, não foi possível reproduzir a mesma curva definida pelo usuário. Outro ponto a ser destacado é que a temperatura não segue fielmente a

curva pré-definida quando a mesma tem uma inclinação elevada, ou seja, nos casos em que uma curva pré-definida exige um aumento muito grande num curto espaço de tempo não é possível obter resultados precisos. Esse fato foi observado no teste da Figura 53, onde foi definido que para o instante de tempo 0 que a temperatura deveria estar em 80 graus Celsius e no instante de tempo de 1 minuto a temperatura deveria estar em 150 graus Celsius. Portanto, para esses casos fica evidente a limitação do aquecimento da pipoqueira. Vale destacar que para a temperatura aumentar de 80 graus até os 170 graus, foram necessários aproximadamente 3 minutos. Os dados deste teste podem ser consultados pelo repositório [online](#).

Mesmo com a limitação de aquecimento da pipoqueira, a Figura 55 mostra a coloração dos grãos de café após o fim do processo de torrefação. Notou-se também que houve uma diminuição da massa do café de 30 para 28 gramas devido a perda de umidade dos grãos.



Figura 54 – Café verde
Fonte: Próprio Autor

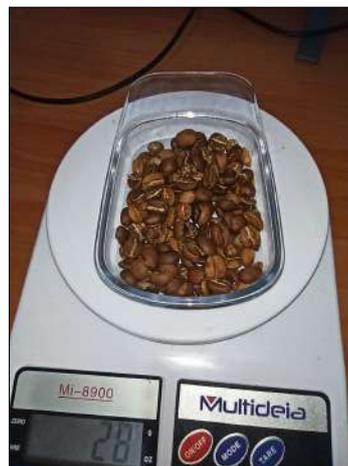


Figura 55 – Café após Torrefação
- “torra 2”
Fonte: Próprio Autor

Outro teste foi realizado com a finalidade de elaborar uma curva de torra que correspondesse à curva definida pelo usuário e dentro da limitação do aquecimento da pipoqueira elétrica. Foi criada então uma nova torra no aplicativo, denominada “torra 3”, com temperatura máxima de 165 graus Celsius.

Para essa nova torra foi utilizado um tempo de torrefação de 12 minutos. Pôde-se acompanhá-la em tempo real, conforme mostra a Figura 56. Além disso, para essa torra os grãos de café obtiveram, a olho nu, o mesmo nível de coloração da torra anterior (Figura 55). Os *logs* do teste também se encontram no repositório [online](#).

Desta forma, notou-se que os sinais de controle PID lidaram bem para atuar nos circuitos de resistência e ventilação, pois conforme mostrado pela Figura 56 o sistema responde bem a curva definida pelo usuário.



Figura 56 – Acompanhamento em tempo real - “torra 3”
Fonte: Próprio Autor

Também pôde-se notar que embora a pipoqueira elétrica apresente a limitação de aquecimento, ao realizar uma curva de torra mais longa, ou seja, com maior tempo de torra, o resultado da torrefação dos grãos também é satisfatória.

4.3 Custos Diretos do Protótipo

Foram calculados os custos diretos do desenvolvimento do protótipo, isto é, o preço dos equipamentos utilizados para elaborar o protótipo. Por isso, os custos de desenvolvimento do aplicativo, bem como toda a parte desenvolvimento de software não foi levada em consideração. Os custos diretos estimados foram obtidos através de consultas na internet em julho de 2022 e foram listados na tabela abaixo:

Tabela 3 – Custo Estimado do Protótipo

Itens	Preço
Módulo Leitor Temperatura Max6675 Termopar Tipo K Arduino	R\$ 31.50
Relé de Estado Sólido SSR-40 DA SSR 40 DA 40A FOTEK	R\$ 30.90
Dissipador De Calor de Alumínio Para Relé De Estado Sólido	R\$ 24.90
Pasta Térmica 15g Implastec Processador Cpu	R\$ 4.90
Conector Plug P4 Femea P	R\$ 2.00
Protoboard Breadboard 400 Pontos Furos Pinos	R\$ 8.90
Módulo Rele 1 Canal 5v/10a Com Led Indicador Serve Arduino	R\$ 6.90
Fonte De Alimentacao Estabilizada 24V 2A Bivolt AC/DC	R\$ 15.90
Pipoqueira Britânia Pop Time PIP B-02 - Branco	R\$ 109.90
Cabo Wire Jumper 20cm 40 Fios Femea-macho Protoboard Arduino	R\$ 8.90
Esp32 Wroom Wifi Bluetooth Devkit	R\$ 33.00
Fita Isolante Foxlux 20mx19m Preto Antichamas e Flexível	R\$ 6.58

Itens	Preço
	R\$ 284,28

Para os custos não foram considerados as ferramentas como alicate, furadeira e brocas e equipamentos de solda eletrônica. Também não foram considerados os materiais usados para a estética final do protótipo.

Desta forma, conforme observado na Tabela 3, o custo estimado direto do protótipo foi de R\$ 284,28.

4.4 Comparativo com Equipamentos Similares no Mercado

Um dispositivo de torrefação que pode ser usado em grau de comparação é o IKAWA Home Roasting System. Esse equipamento utiliza a mesma técnica Air Roasting e permite ao usuário torrar grãos de café verde utilizando um aplicativo de celular para os sistemas operacionais Android e IOS. O seu preço atual (Julho de 2022) no [site oficial](#) é de £1,270.00.

Segundo as especificações técnicas encontradas no [site oficial](#), o equipamento (Figura 57) permite uma torrefação de aproximadamente 50 gramas de café por torra. Sua potência máxima é de 1700W e tem massa de 5Kg. Além disso, o IKAWA Home Roasting System possui um design de alumínio extrudado sob medida que oferece excelentes propriedades térmicas e de resistência pensada exclusivamente para melhorar o fluxo de ar quente. Possui também um sistema integrado de coletor de cascas do café que evita a sujeira do ambiente.

No aplicativo o usuário pode definir as curvas de torras que podem chegar até 250 graus Celsius.



Figura 57 – IKAWA Home Roasting System

Fonte: <https://ikawahome.com>

Também é possível acessar torras em destaque dentro do próprio aplicativo e realizar o acompanhamento em tempo real da mesma. A conexão do equipamento com o aplicativo se dá por meio do *Bluetooth* 5.0.

Em grau de comparação em relação ao protótipo deste trabalho, nota-se que a potência máxima da pipoqueira elétrica é de 1200W, permitindo temperaturas até aproximadamente 170 graus Celsius. Também não há um sistema integrado de coletores de cascas de café. Entretanto, os dois equipamentos utilizam o método Air Roasting para a torrefação dos grãos de café.

O comparativo referente a parte de conectividade se difere, uma vez que no presente trabalho foi desenvolvido através do protocolo TCP/IP e o IKAWA trabalha baseado na tecnologia *Bluetooth*. Em relação ao aplicativo, ambos assemelham-se em devido à possibilidade de criação de curvas de torras específicas bem como acompanhá-las em tempo real. Entretanto neste projeto assumiu-se que o aplicativo e a torradeira estarão sempre conectadas à internet enquanto o IKAWA, por utilizar a tecnologia *Bluetooth*, deve estar próximo ao dispositivo móvel.

5 Conclusões

O presente trabalho apresentou um protótipo de torrador de café de baixo custo ao adaptar uma pipoqueira elétrica com finalidade de realizar a torrefação de grãos de café verde utilizando a técnica Air Roasting. O projeto realizado utilizou-se da teoria de controle realimentados para lidar com o controle dos circuitos de aquecimento e ventilação de uma pipoqueira elétrica. Desta forma, foi possível implementar um controle clássico PID discreto utilizando o microcontrolador ESP32 e também foi possível realizar testes de sintonia para que os sinais deste controlador atuassem de forma adequada nos circuitos da pipoqueira.

Sob aspectos da IoT, a adaptação da pipoqueira elétrica com sensor de temperatura e um microcontrolador ESP32 possibilitou à pipoqueira sua identificação única e a comunicação de dados à internet. Para a aplicação, foi desenvolvido um aplicativo móvel que possibilitou a criação de multiusuários que podem arquitetar suas próprias torras, bem como compartilhá-las e acompanhá-las em tempo real durante o processo de torrefação do café.

No decorrer do projeto, foram encontrados diversos desafios e obstáculos para a criação do protótipo. A montagem do protótipo foi considerada um grande desafio do projeto devido a complexidade. Essa montagem consistiu em abrir a pipoqueira elétrica, identificar e separar os circuitos de aquecimento e ventilação e uni-los ao circuito completo. Além disso, os desafios relacionados à programação do microcontrolador foram grandes, uma vez que lidar com comunicações assíncronas causavam erros e travamentos no microcontrolador. Outro desafio foi a criação de um aplicativo para dispositivos móveis Android que pudessem suportar multiusuários, identificar dispositivos na rede e o principal que foi acompanhar a torrefação de café em tempo real.

Ao final do projeto foram identificados alguns pontos positivos bem como pontos negativos referentes ao aplicativo para dispositivos móveis. Um dos pontos positivos é que ele permite ao usuário realizar operações básicas como criar, ler, editar e excluir suas próprias torras. Além disso, é possível ao usuário acessar torras públicas de outros usuários, configurando-se como um sistema de torras compartilhadas entre usuários. Ademais, o fato de o cadastro da torradeira poder ser feito de duas formas (uma via *QRCode* e a outra listando os dispositivos na rede) possibilita que o *smartphone* esteja ou não na mesma rede local. Dentre os pontos positivos citados para o aplicativo, o destaque é a tela de acompanhamento da torra em tempo real. Essa tela permite a visualização completa do processo de torrefação do café pelo usuário.

Os pontos negativos encontrados em relação ao aplicativo foram a total dependência

de conexão com a Internet, visto que todos os processos carregados pelo aplicativo necessitam de acesso ao Firebase. Por este motivo, caso ocorra uma perda de conexão com a mesma, o aplicativo e dispositivo não atuarão de forma correta.

Assim como para o aplicativo, foram considerados pontos positivos e negativos em relação ao protótipo da torradeira de café. O principal ponto positivo está no fato de simplesmente ser possível torrar o café. O modelo Air Roasting, apesar de pouco difundido, vem ganhando espaço pela simplicidade de se realizar uma torra de café. Também foi considerado um ponto positivo a portabilidade da pipoqueira, uma vez que ela é leve (cerca de 1Kg). Outro ponto positivo identificado referente ao protótipo deve-se ao fato da carcaça do equipamento esquentar pouco durante a torrefação. Além disso, o fato de o custo estimado da confecção protótipo ser inferior a R\$ 300.00, tornou-se um fator importante a ser destacado.

Sob os aspectos negativos encontrados para o protótipo, o principal se refere a limitação do circuito de aquecimento, visto que a maior temperatura medida pelo sensor foi de 171 graus Celsius. Por isso, as curvas de torras definidas com valores de temperaturas acima disso não podem ser acompanhadas. Além disso, outro ponto negativo se refere ao circuito de ventilação da pipoqueira uma vez que o motor DC gera uma ventilação incapaz de agitar os grãos de café em grandes quantidades. Assim, o protótipo consegue suportar até no máximo 40 gramas de café verde, mantendo a rotatividade dos mesmos. Também considerado um ponto negativo, é a falta de um coletor de cascas de café, visto que durante a torra as películas de café são despejadas para fora da pipoqueira, sujando o ambiente. Por fim, outro ponto negativo se dá quando a torra é finalizada, sendo necessário virar o protótipo de ponta-cabeça para que os grãos sejam retirados da parte interna da pipoqueira.

Por fim, conclui-se que de modo geral o protótipo desenvolvido teve um resultado satisfatório no que se refere à torrefação de café. Além disso, os conceitos de IoT e controle realimentado possibilitaram que a pipoqueira elétrica se tornasse um equipamento inteligente e controlável via internet. O trabalho também foi importante para despertar o interesse em aplicações e inovações IoT, que melhoram e facilitam a vida das pessoas, e também fortalecem o empreendedorismo no Brasil.

5.1 Trabalhos Futuros

Com o objetivo de realizar melhorias para o projeto desenvolvido neste trabalho, alguns pontos foram levantados para esse propósito. Notou-se que, embora a pipoqueira elétrica permita realizar torras de café, ela possui limitações de aquecimento e ventilação. Por isso, o ideal seria a substituição da pipoqueira elétrica pela criação de um protótipo mais robusto, isto é, um equipamento que permita um aquecimento a temperaturas

próximas a 250 graus Celsius, fazendo com que as torras tenham um menor tempo de duração.

Além disso, a criação de um novo protótipo com um motor de rotação mais forte poderia propiciar uma ventilação mais potente e conseqüentemente mais grãos de café poderiam ser torrados em uma única vez. Ademais, seria importante para esse novo protótipo poder lidar com a coleta das cascas de café expelidas durante a torrefação. Também seria interessante resolver o fato de ter que virar de ponta-cabeça para realizar a retirada do café, uma vez que isso é um problema existente na pipoqueira.

Para o aplicativo, seria importante aumentar os meios de comunicação para permitir que o aplicativo e o equipamento consiga realizar torras sem estar necessariamente conectadas à internet. Por isso, adicionar a tecnologia *bluetooth* em paralelo com a já existente seria muito viável para melhoria do produto final.

Além disso, para uma melhora na sintonia do controlador, seria ideal fazer a modelagem do sistema a fim de obter uma função de transferência e realizar simulações em software para conseguir otimizar os parâmetros do controle PID. Outra abordagem envolveria obter um modelo a partir dos *logs* das torras e através de técnicas de aprendizado de máquina seria possível obter um modelo para modelar os circuitos de aquecimento e ventilação. A partir disso, comparar esses modelos através de resultados qualitativos poderiam melhorar o controle do processo de torra.

5.2 Material Complementar

Nesta seção estão disponíveis os links referentes às etapas do desenvolvimento e validações da proposta do projeto:

- Desenvolvimento da programação para o hardware: [Github](#).
- Desenvolvimento do aplicativo Android: [Github](#)
- Dados dos testes de torrefação, realizados nas seções 4.2.1 e 4.2.2, encontram-se junto ao projeto do aplicativo Android, na pasta denominada logs-torras: [Github](#).

Referências

- AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys Tutorials*, v. 17, n. 4, p. 2347–2376, 2015. Citado na página 20.
- ALMEIDA, H. Apresentação, internet das coisas, tudo conectado. *Sociedade Brasileira de Computação*, Disponível em: <https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa_29_pdf/comp_brasil_2015_4.pdf>, 2015. Citado na página 14.
- ANDRADE, H. C. C. Atribuição de sentidos e agregação de valor: insumos para o turismo rural em regiões cafeeicultoras. *Revista Brasileira de Ecoturismo*, São Paulo, v. 8, n. 2, p. 333–346, 2015. Citado na página 15.
- BORGIA, E. The internet of things vision: Key features, applications and open issues. *Computer Communications*, v. 54, p. 1–31, 2014. ISSN 0140-3664. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0140366414003168>>. Citado na página 23.
- BORRELL, I.; MATAIX, C.; CARRASCO-GALLEGO, R. R. smallholder farmers in the speciality coffee industry: opportunities, constraints and the businesses that are making it possible. *IDS Bulletin*, Brighton, v. 46, n. 3, p. 29–44, 2015. Citado na página 15.
- BRASIL, M. da Ciência Tecnologia e I. Plano nacional de internet das coisas - análise de oferta e demanda. Disponível em: <https://www.gov.br/mcti/pt-br/acompanhe-o-mcti/transformacaodigital/arquivosinternetdascoisas/fase1_3c_analise-de-demanda.pdf>, 2017. Citado na página 14.
- COSTA, V. *A nova onda do café e os torradores domésticos*. 2018. <<https://medium.com/@relatos.cafe/a-nova-onda-do-café-e-os-torradores-domésticos-3c03869d0225>>. Citado 2 vezes nas páginas 35 e 39.
- DEITEL, P.; DEITEL, H.; DEITEL, A. *Android: como programar - 2a Edição*. Porto Alegre: Bookman, 2015. Citado 2 vezes nas páginas 31 e 32.
- DELICATO, F.; PIRES, P.; BATISTA, T. *Middleware Solutions for the Internet of Things*. Springer London, 2013. (SpringerBriefs in Computer Science). ISBN 9781447154808. Disponível em: <<https://books.google.com.br/books?id=5YDJngEACAAJ>>. Citado na página 20.
- DORF, R. C.; BISHOP, R. H. *Sistemas de controle modernos. LTC - Livros Técnicos e Científicos Editora Ltda*, 13ª edição, Rio de Janeiro, 2018. Citado 5 vezes nas páginas 14, 25, 27, 29 e 30.
- FERRASI, F. A.; MORGADO, E. M. Internet das coisas: captação autônoma de dados com o uso das tecnologias móveis. *Revista Multiplicidade*, v. 7, n. 7, set. 2018. Disponível em: <<https://revistas.fibbauru.br/multiplicidadefib/article/view/78>>. Citado na página 20.

- FOX, R. W. et al. *Introdução à Mecânica dos Fluidos. 8ª ed.* Rio de Janeiro: Rio de Janeiro, 2014. Citado na página 33.
- FURGERI, S. *Java 7 - Ensino didático - 2a Ed.* São Paulo: São Paulo, 2010. Citado na página 32.
- GAMA, K.; TOUSEAU, L.; DONSEZ, D. Combining heterogeneous service technologies for building an internet of things middleware. *Computer Communications*, v. 35, n. 4, p. 405–417, 2012. ISSN 0140-3664. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0140366411003586>>. Citado na página 18.
- GODOI, M. G. d.; ARAÚJO, L. S. A internet das coisas: evolução, impactos e benefícios. *Revista Interface Tecnológica*, v. 16, n. 1, p. 19–30, jun. 2019. Disponível em: <<https://revista.fatectq.edu.br/interfacetecnologica/article/view/538>>. Citado na página 18.
- GOLNARAGHI, F.; KUO, B. C. Sistemas de controle automático. *LTC - Livros Técnicos e Científicos Editora Ltda*, 9ª edição, Rio de Janeiro, 2012. Citado 2 vezes nas páginas 28 e 29.
- GOOGLE. *Firestore Realtime Database: Store and sync data in real time.* 2022. <<https://firebase.google.com/products/realtime-database/>>. Citado na página 23.
- HELDT, R. Relé de estado sólido: funcionamento e vantagens. Disponível em: <<https://blog.rhmateriaiseletricos.com.br/como-funciona-um-rele-de-estado-solido/>>, 2019. Citado na página 38.
- KESTERING, D. A. Simulação fluidodinâmica de um leito fluidizado empregando correlações de arrasto gás-sólido ajustadas por valores experimentais. *Universidade do Vale do Rio dos Sinos*, São Leopoldo, 2016. Citado na página 33.
- LEE, V.; SCHNEIDER, H.; SCHELL, R. *Aplicações móveis: arquitetura, projeto e desenvolvimento.* [S.l.]: São Paulo: Pearson, 2005. Citado 2 vezes nas páginas 30 e 31.
- MASCHIETTO, L. G. et al. Arquitetura e infraestrutura de iot. *Grupo A*, 1ª Edição, Porto Alegre: SAGAH, 2020. Citado 2 vezes nas páginas 20 e 22.
- MATTEDE, H. O que são sensores e quais as suas aplicações?. Disponível em: <<https://www.mundodaeletrica.com.br/o-que-sao-sensores-e-quais-as-suas-aplicacoes/>>, 2018. Citado na página 26.
- McKinsey Global Institute. The internet of things : mapping the value beyond the hype. Disponível em: <<https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/the-internet-of-things-the-value-of-digitizing-the-physical-world>>, 2015. Citado na página 23.
- MELO, W. L. B. A importância da informação sobre do grau de torra do café e sua influência nas características organolépticas da bebida. *Embrapa Instrumentação Agropecuária*, São Paulo, 2004. Citado na página 15.
- MOTT, J. *A guide to fluid bed coffee roasters.* 2021. <<https://perfectdailygrind.com/2021/05/a-guide-to-fluid-bed-coffee-roasters/>>. Citado 2 vezes nas páginas 15 e 33.

SANTOS, B. P. et al. Internet das coisas: da teoria à prática. *Departamento de Ciência da Computação, Universidade Federal de Minas Gerais (UFMG)*, Belo Horizonte, MG, Brasil, 2016. Citado 3 vezes nas páginas 18, 19 e 20.

SCHENKER; STEFAN. Investigations on the hot air roasting of coffee beans. *ETH Zürich*, Disponível em: <<https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/144592/eth-23461-01.pdf>>, 2000. Citado na página 33.

SIMAS, V. L.; BORGES, O. T.; COUTO, J. M. C. *Desenvolvimento para dispositivos móveis - Volume 2*. [S.l.]: Porto Alegre: SAGAH, 2019. Citado na página 31.

SMITH, C. A.; CORRIPIO, A. B. Princípios e prática do controle automático de processo. *LTC - Livros Técnicos e Científicos Editora Ltda*, 3ª edição, Rio de Janeiro, 2012. Citado 5 vezes nas páginas 25, 26, 27, 28 e 30.

TRAFANIUC, V. *Saiba o que é Google Cloud Platform e quais suas vantagens*. 2021. <<https://maplink.global/blog/o-que-e-google-cloud/>>. Citado na página 23.