

PROTÓTIPO DE SISTEMA DE AUTOMAÇÃO RESIDENCIAL INTEGRADO COM RASPBERRY PI UTILIZANDO WINDOWS

Aluno: Plamedi L. Lusembo

Orientador: Prof. Miguel Alexandre Wisintainer

Roteiro

- **Introdução**
- **Objetivos**
- **Fundamentação Teórica**
- **Trabalhos Correlatos**
- **Requisitos**
- **Especificação**
- **Implementação**
- **Operacionalidade da Implementação**
- **Resultados e Discussões**
- **Conclusões e Extensões**

Introdução

- **A IoT estimulou a inovação:**
 - a) **na criação de plataformas de prototipagem eletrônico;**
 - b) **e também no desenvolvimento de softwares e sistemas operacionais compatíveis.**

- **A compatibilidade do Windows com o Raspberry Pi ampliou as possibilidades de prototipagem para o desenvolvimento IoT.**

Objetivos

- **Implementar um protótipo de um sistema de controle e automação residencial baseado no Raspberry Pi 3 Model B com o Windows 10 IoT Core.**

Objetivos

- **Pesquisar sobre como utilizar o Raspberry Pi 3 Model B com o Windows 10 IoT Core;**
- **Construir um hardware baseado no Raspberry Pi 3 Model B para servir de central de controle;**
- **Desenvolver o aplicativo gerenciador;**
- **Desenvolver um software embarcado para rodar no Raspberry Pi 3 Model B como servidor.**

Fundamentação Teórica

Assuntos que fundamentam a pesquisa realizada e o desenvolvimento do protótipo:

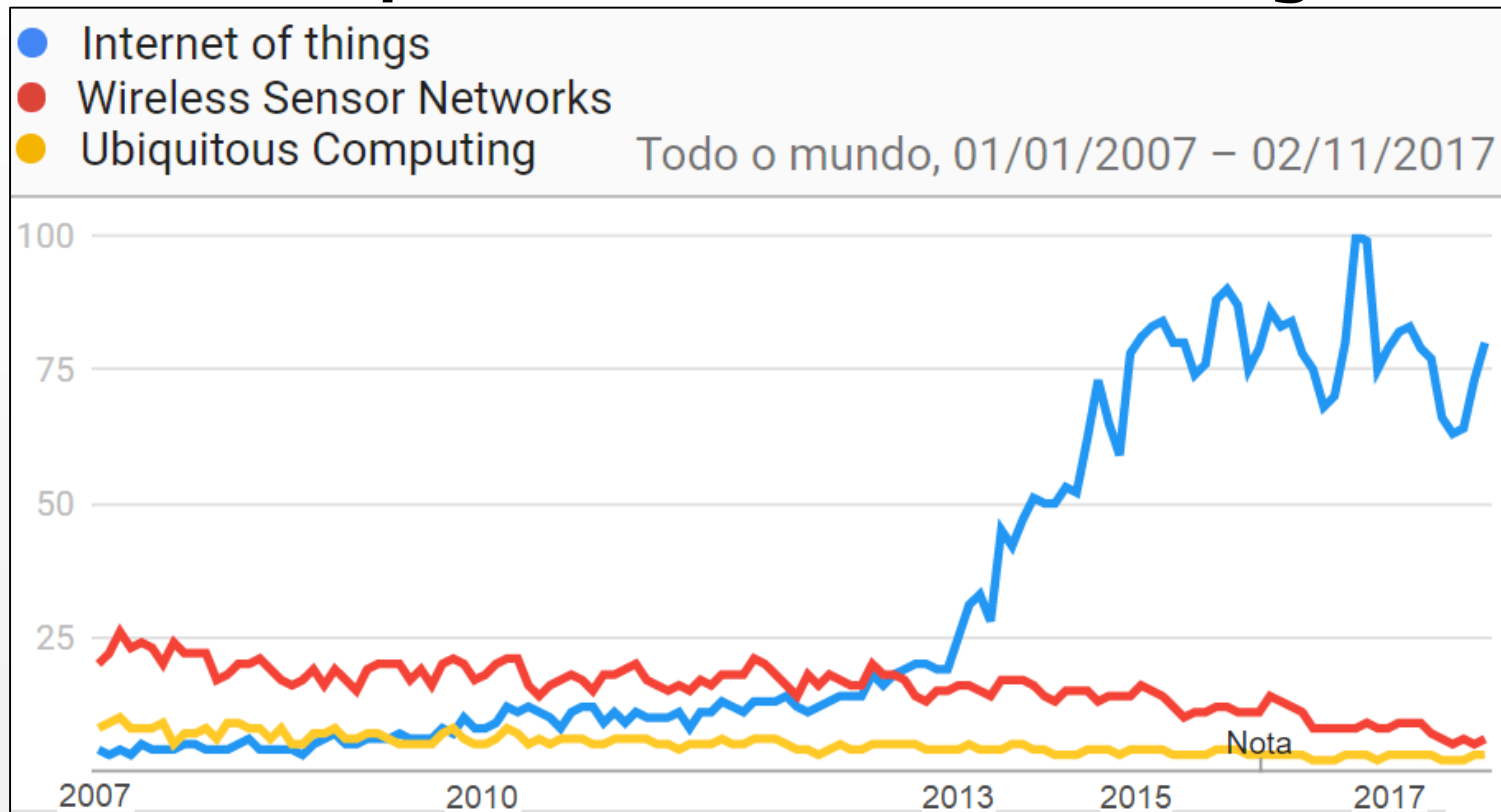
- IoT;**
- Raspberry Pi;**
- Windows IoT;**
- Automação.**

• IoT

- Kevin Ashton é reconhecido por usar o termo 'Internet of Things' pela primeira vez em 1999;
- Visa tornar os equipamentos tradicionalmente não-conectáveis capazes de serem identificados no meio digital e conectados à rede;
- Uso de sensores, atuadores e controladores.

Fundamentação Teórica

Aumento considerável da frequência de busca pelo termo Internet of Things



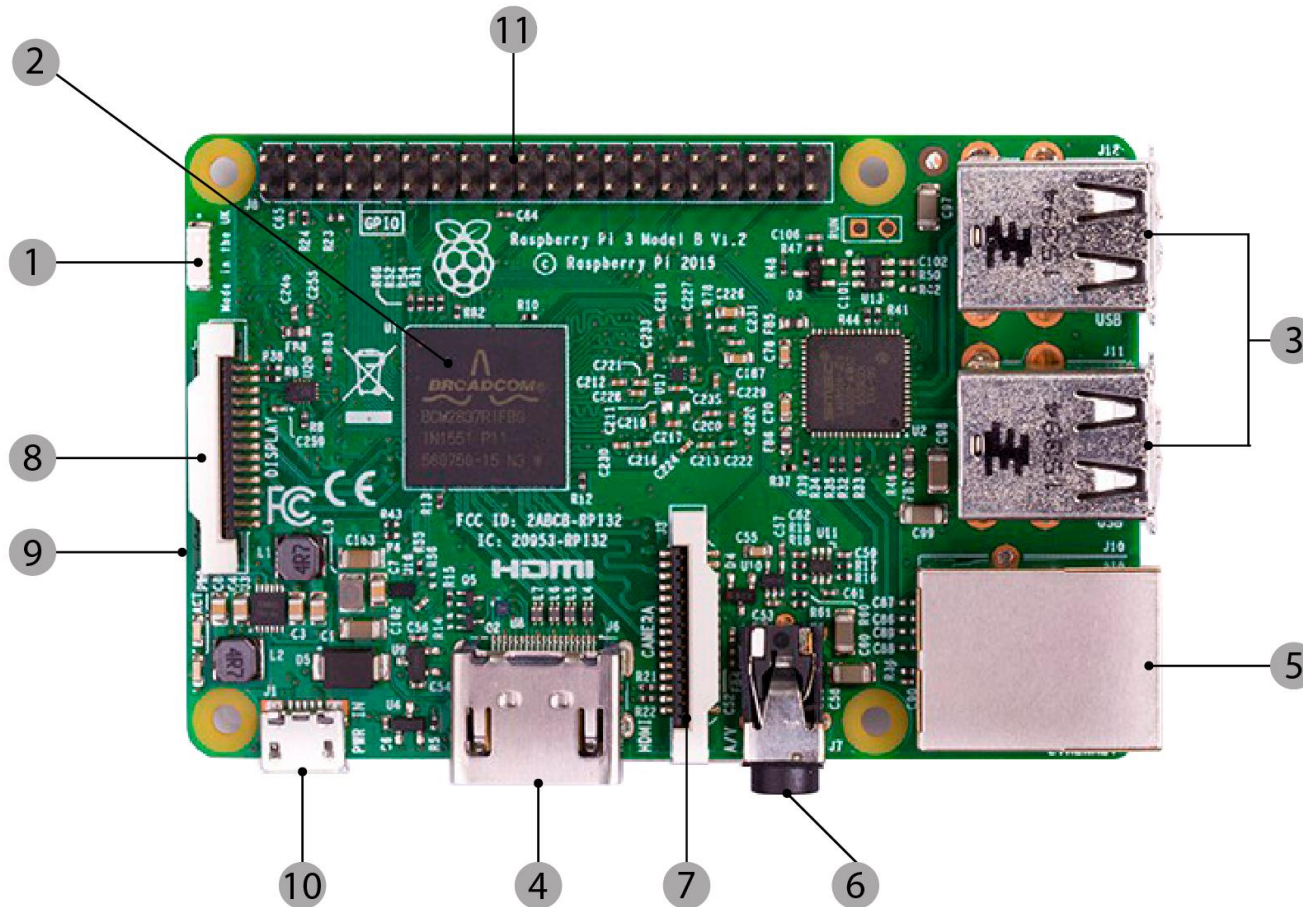
Fonte: elaborado pelo autor.

• Raspberry Pi

- Minicomputador de placa única, de baixo custo e do tamanho de um cartão de crédito;
- Concebido principalmente para promover o aprendizado da programação de computadores nas escolas.
- Os primeiros modelos foram lançados no ano 2012;

Fundamentação Teórica

Raspberry Pi 3 Model B



Fonte: adaptado do Raspberry PI Foundation (2017).

• **Windows IoT**

- Conjunto de SO da Microsoft para dispositivos pequenos com ou sem monitor;
- É executado em dispositivos ARM e x86/x64”;
- Sucessor do Windows Embedded.

Windows 10 IoT Core: Versão compacta do Windows 10 projetada e otimizada para dispositivos embarcados.

Windows 10 IoT Core

Características principais:

- interação direta com o hardware;
- possibilidade de executar sistemas sem monitor ou display;
- integração da API Universal Windows Platform (UWP).

Fundamentação Teórica

UWP

- filosofia “one Windows”;
- possibilita que software desenvolvido para Windows 10 seja executado em diferentes dispositivos do Windows 10:



Fonte: Microsoft (2017).

• Automação

Todo processo que:

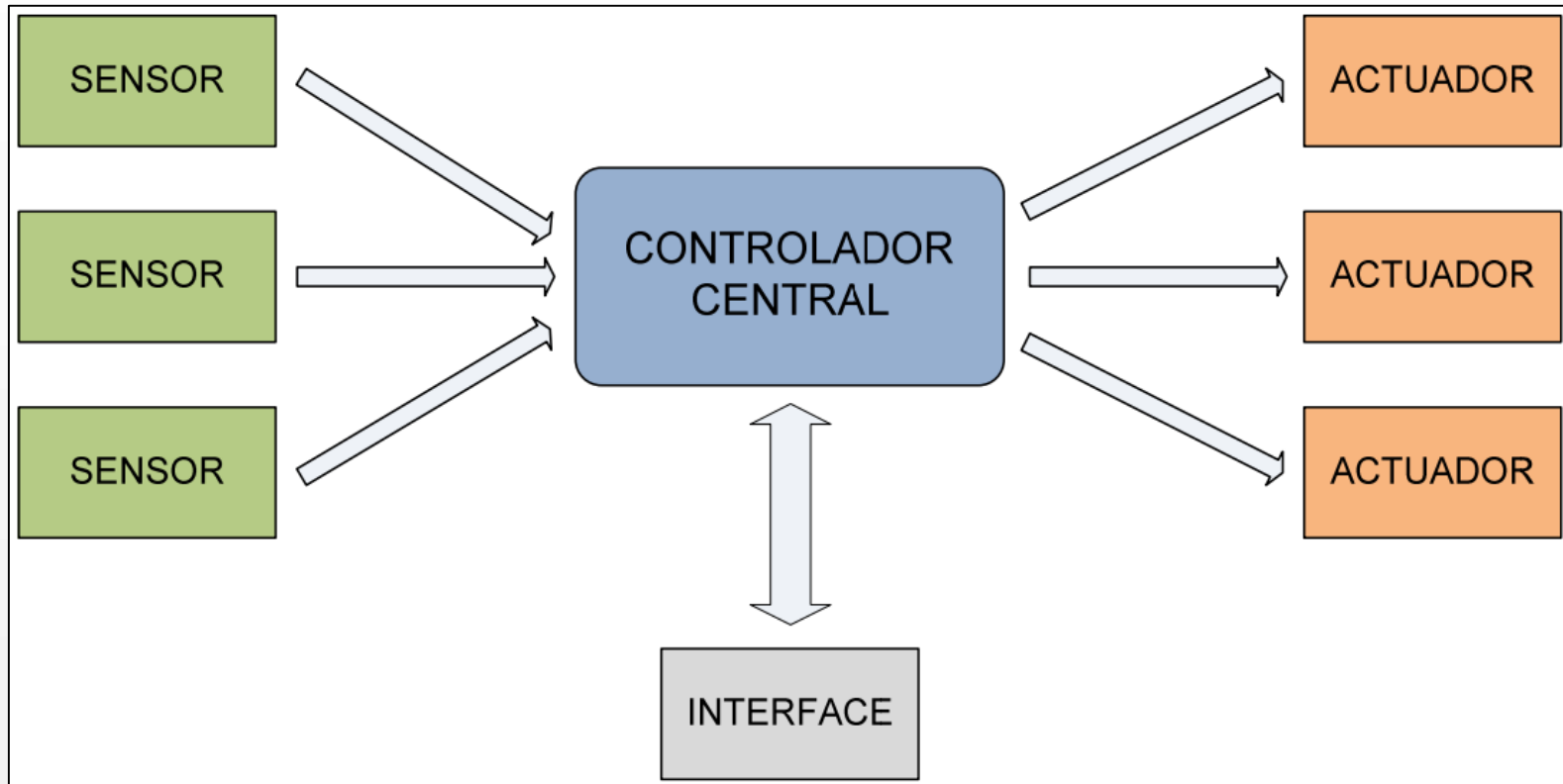
- realiza tarefas e atividades de “forma autônoma”;
- auxilia o homem em suas tarefas do dia-a-dia.

Automação Residencial

- Aplicação da automação em residências;
- Também conhecida como **Domótica**.

Fundamentação Teórica

Arquitetura centralizada de um sistema de automação residencial



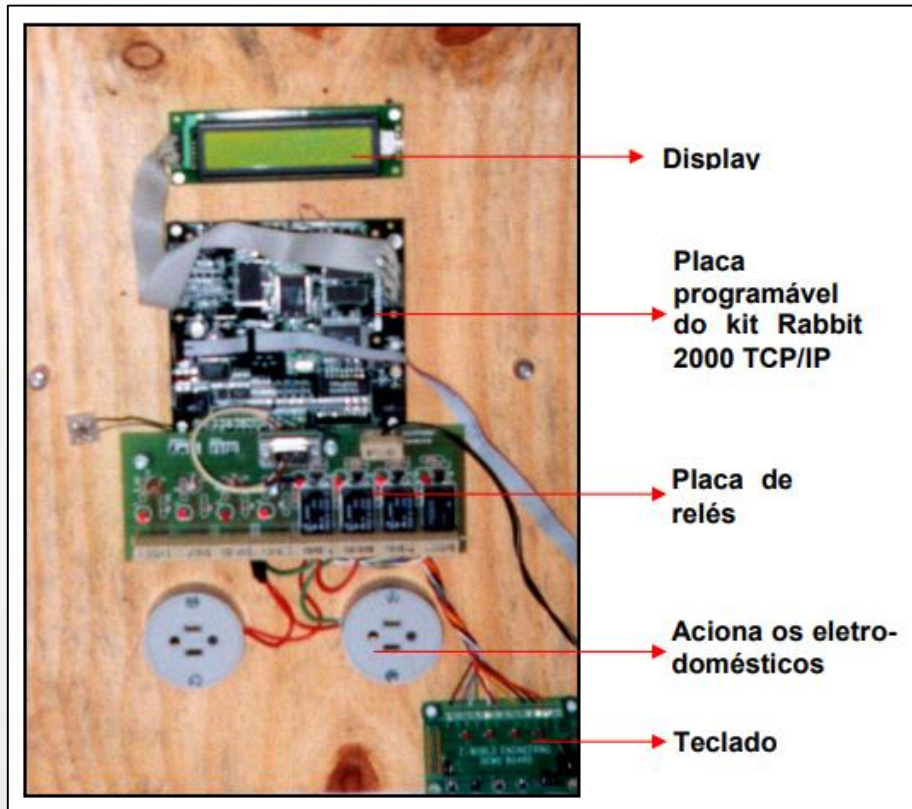
Fonte: Ferreira (2008).

Trabalhos Correlatos

3 trabalhos com características similares aos principais objetivos do trabalho desenvolvido foram consultados.

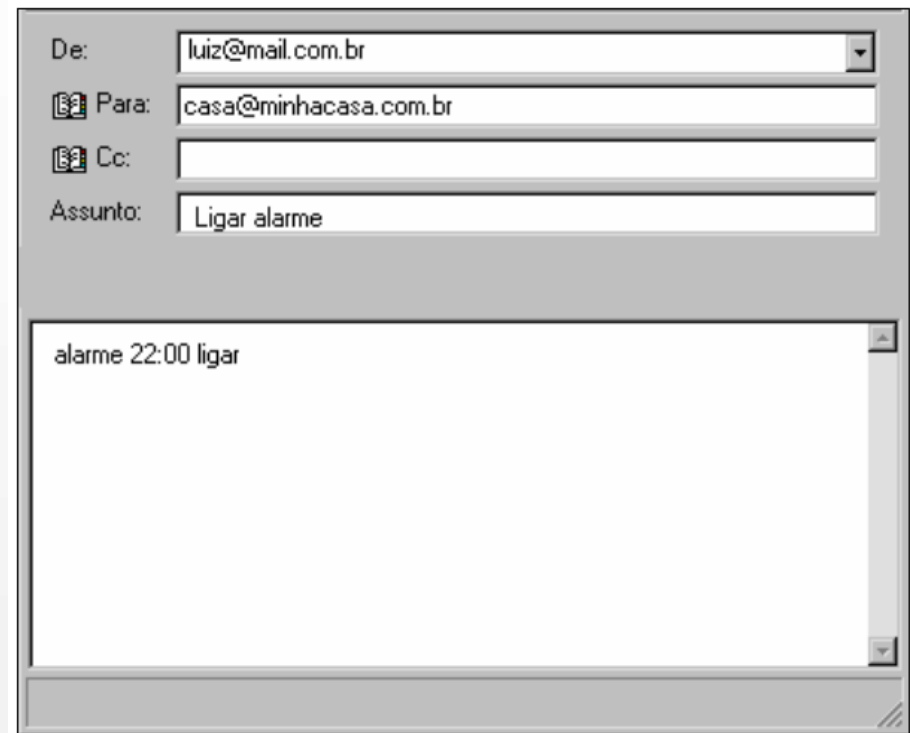
- **Sistema para automação e controle residencial via e-mail**

Hardware do Sistema



Fonte: Censi (2001).

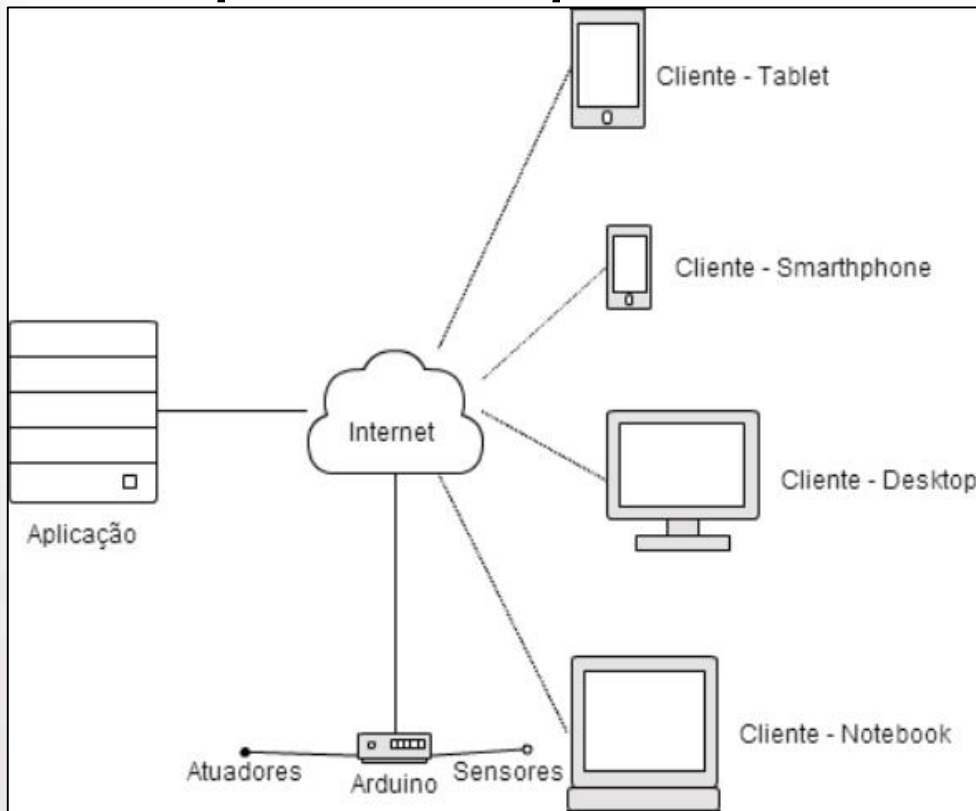
E-mail com tarefa agendada



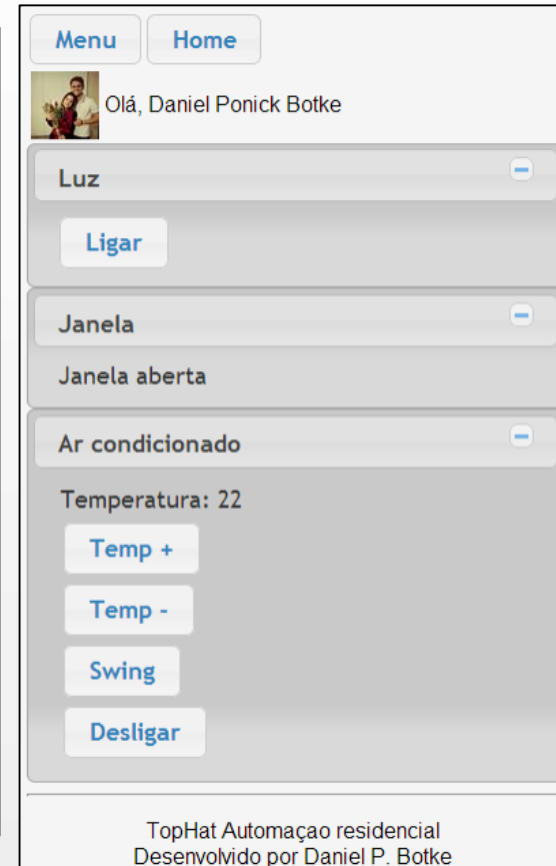
Fonte: Censi (2001).

- **Automação de residência através de aplicação integrada com arduino**

Arquitetura da aplicação



Tela de um cômodo da casa



Fonte: Botke (2014).

Fonte: Botke (2014).

Trabalhos Correlatos

- **Sistema para automação e controle residencial via twitter**

Envio de mensagens pela página oficial do Twitter

Placa FEZ Domino

Entradas e saídas digitais

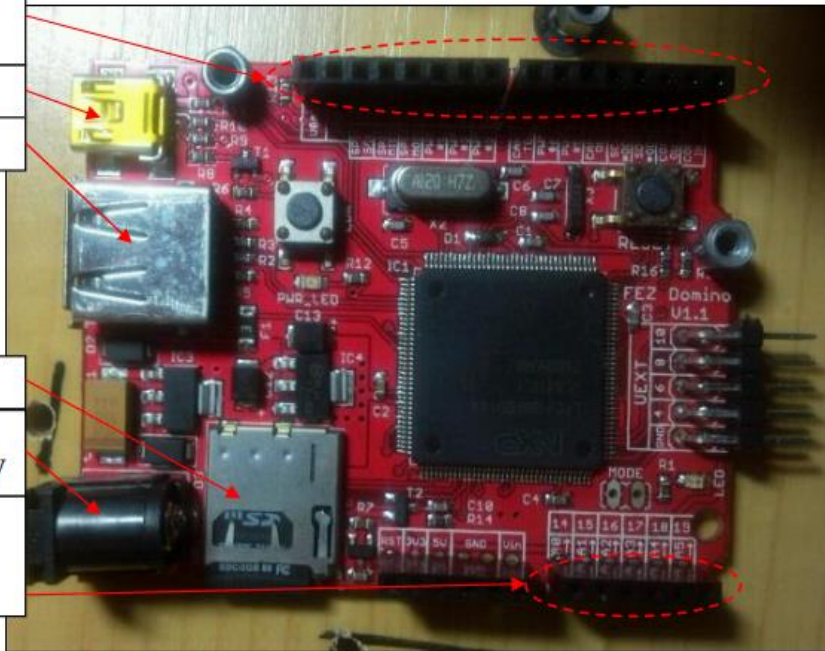
Mini USB

USB Host

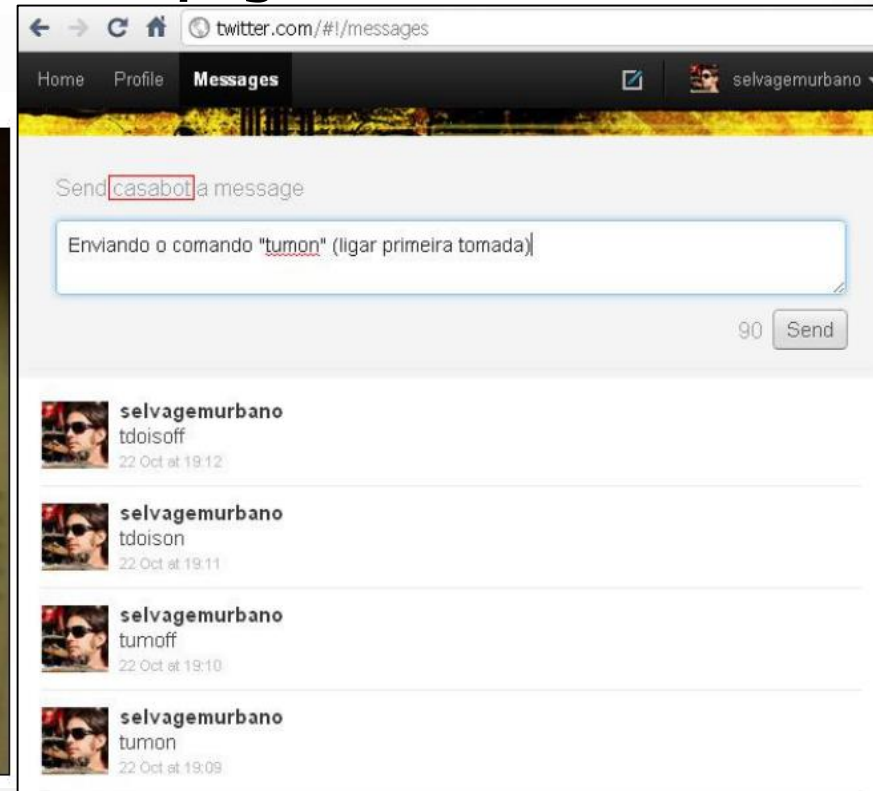
MicroSD

Alimentação externa 6 a 12V

Entradas e saídas analógicas



Fonte: Gadotti (2010).



Fonte: Gadotti (2010).

Requisitos

Requisitos Funcionais e Não Funcionais

• **Requisitos Funcionais**

- **dispor de um mecanismo de autenticação;**
- **permitir que o usuário altere as configurações de conexão;**
- **possibilitar o controle da iluminação;**
- **possibilitar ligar e desligar o ar condicionado;**
- **possibilitar a consulta da temperatura e umidade;**

• **Requisitos Funcionais (Continuação)**

- possibilitar a abertura e o fechamento do portão;
- possibilitar a ativação e a desativação do alarme;
- permitir o envio automático de e-mail de notificação sobre o disparo do alarme;
- possibilitar a visualização do histórico de disparos do alarme;
- possibilitar a visualização do status dos equipamentos.

• **Requisitos Não Funcionais**

- possibilitar que o aplicativo gerenciador seja compatível com o sistema operacional Android;
- utilizar ícones intuitivos na interface gráfica do usuário no aplicativo gerenciador;
- utilizar sockets TCP/IP para a comunicação entre o aplicativo gerenciador e o software embarcado no Raspberry Pi 3 Model B;

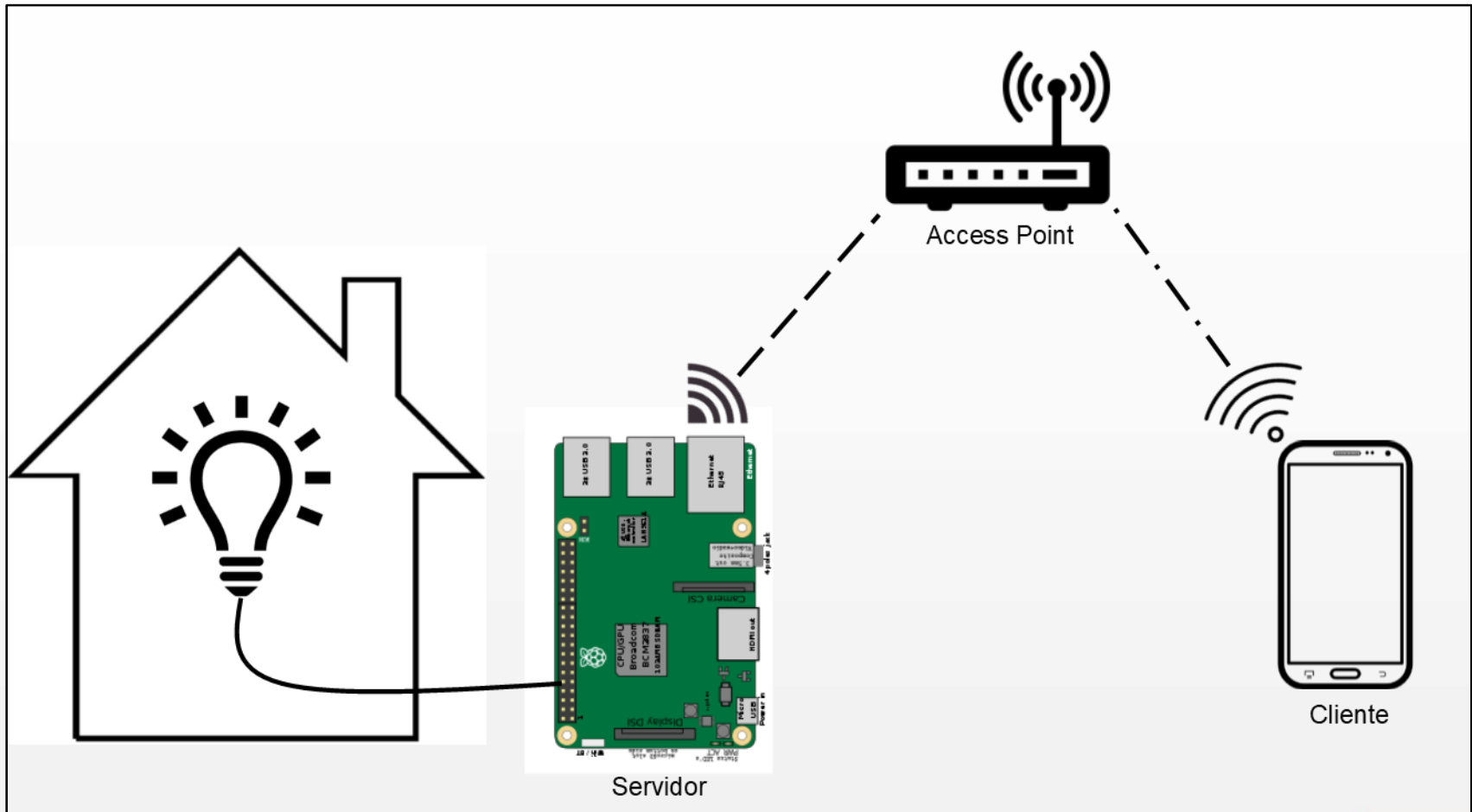
- **Requisitos Não Funcionais (Continuação)**
 - garantir um tempo de resposta com timeout de no máximo 20 segundos para a execução das tarefas;
 - utilizar o sistema de gerenciamento de banco de dados SQLite;
 - ser implementado em C# utilizando o Visual Studio para o software embarcado;
 - ser implementado em Java utilizando o Android Studio para o aplicativo gerenciador.

Especificação

Diagramas de distribuição, de casos de uso e de atividades

Especificação

Diagrama de distribuição do sistema



Fonte: elaborado pelo autor.

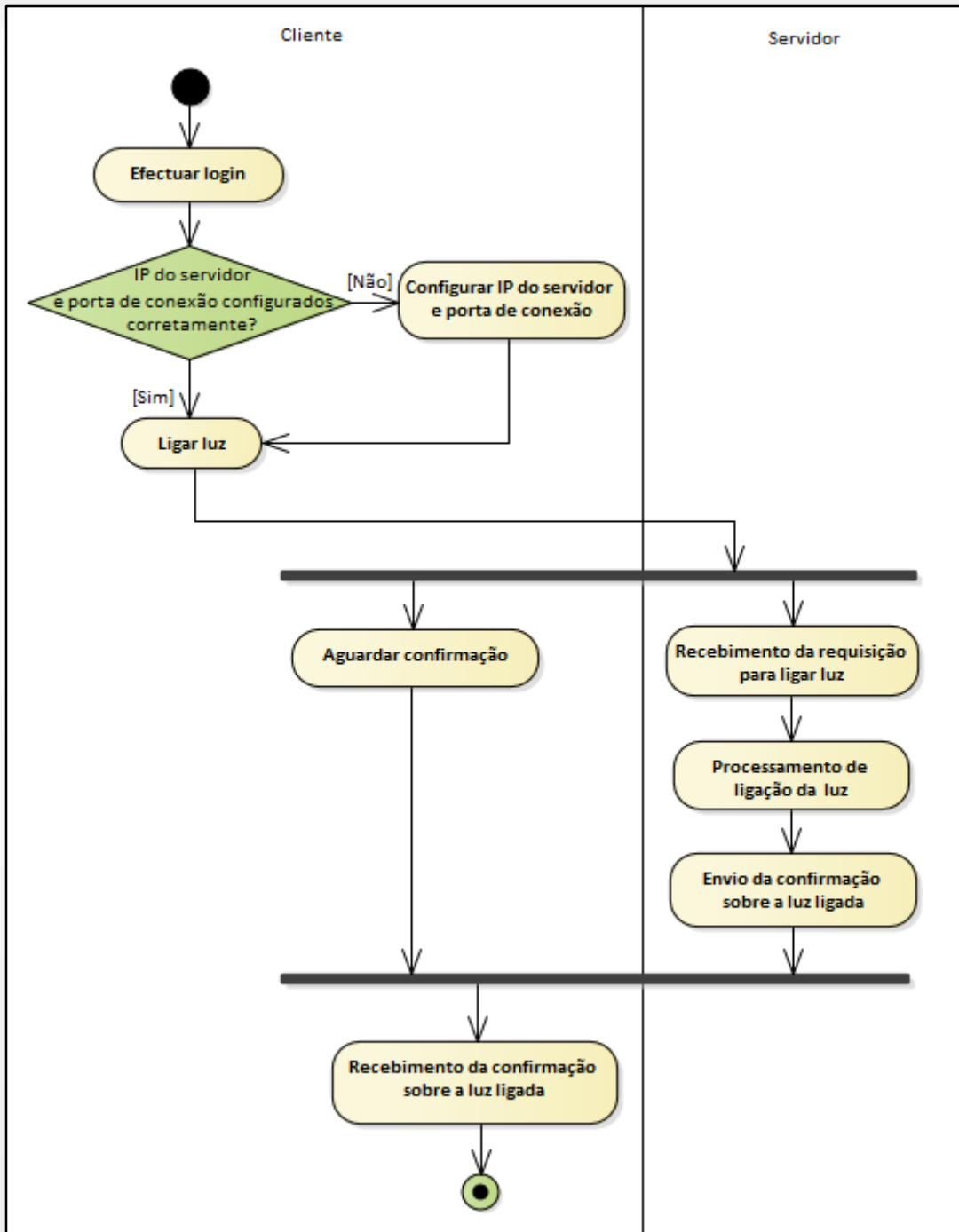
Especificação

Diagrama de casos de uso



Especificação

Diagrama de atividades



Implementação

Técnicas e ferramentas

1. Hardware do sistema

Hardware do sistema

Kit CanaKit Raspberry Pi 3 Starter Kit

CanaKit Raspberry Pi 3 Starter Kit

Model B | 1 GB RAM | 1.2 GHz | Quad-Core CPU



- > Credit-Card Sized Computer
- > Turn your TV into a Media Center
- > Learn to Code & Explore Computing
- > Built-In Wifi & Bluetooth



KIT INCLUDES RASPBERRY PI 3 AND ...

PREMIUM CASE



2.5A POWER ADAPTER



32 GB CLASS 10 MICROSD CARD



PRE-LOADED
WITH OPERATING SYSTEM

MICROSD USB READER



HEAT SINKS



PREMIUM HDMI CABLE



WiFi & BLUETOOTH

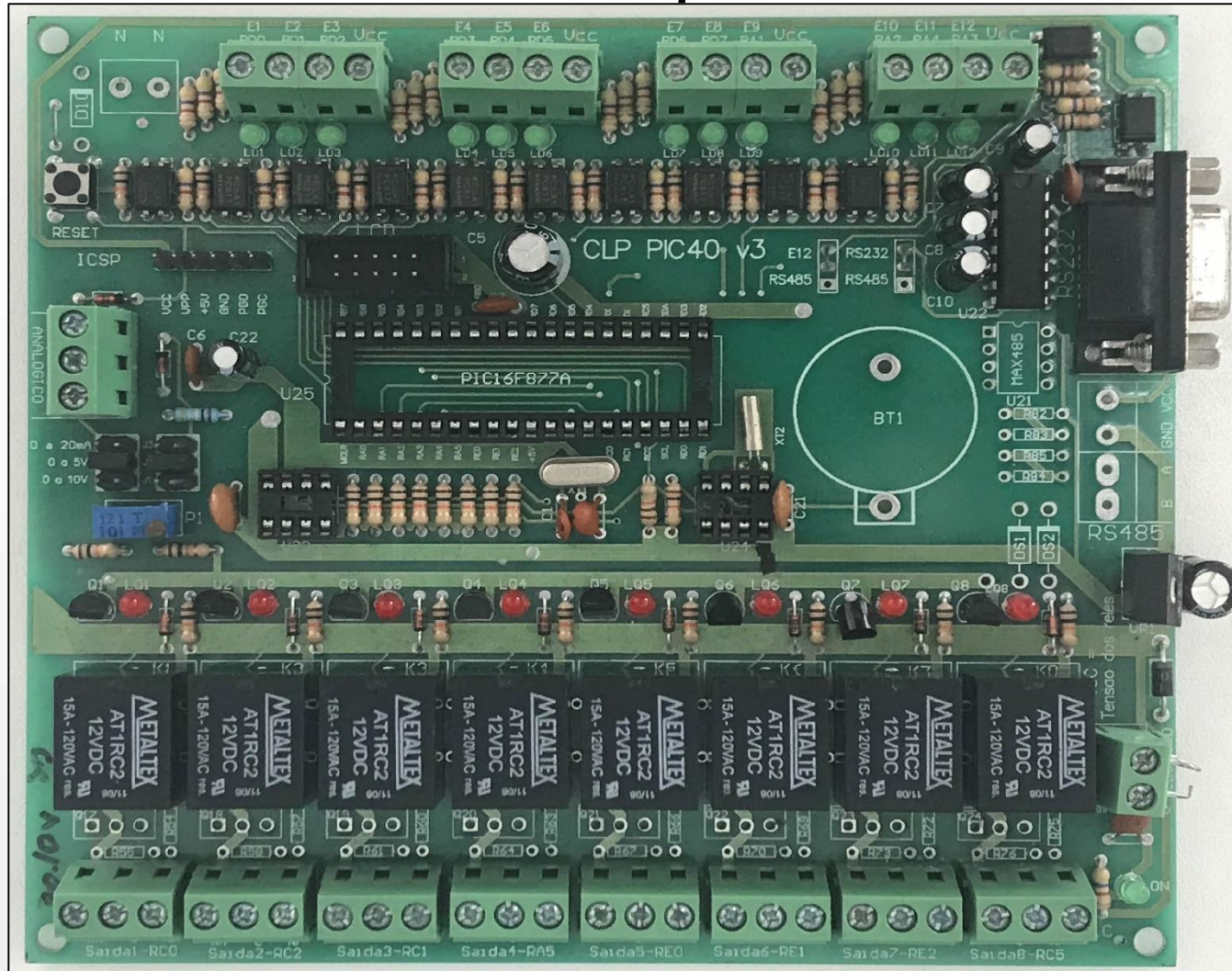


QUICK-START GUIDE



Hardware do sistema

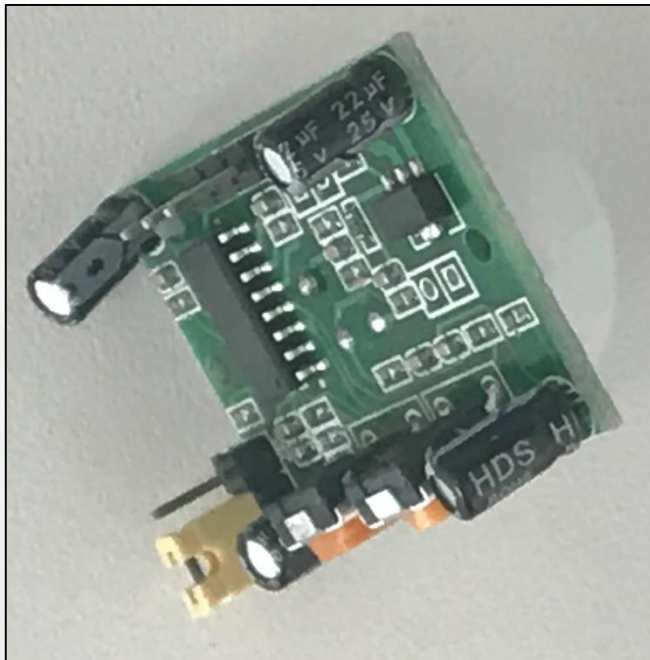
Módulo relé Clp Pic40-v4



Fonte: elaborado pelo autor.

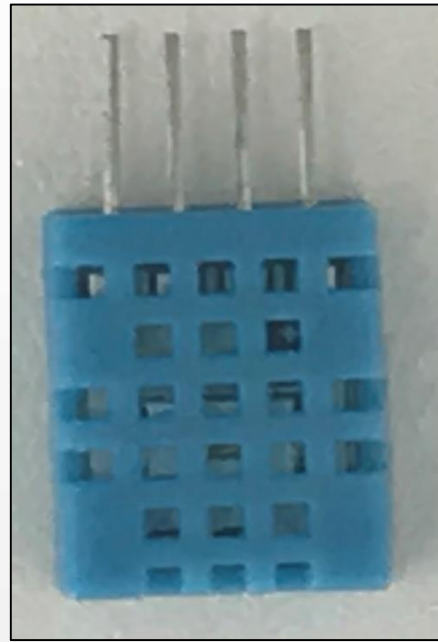
Hardware do sistema

**Módulo sensor de movimento
PIR - HC-SR501**



Fonte: elaborado pelo autor.

**Módulo sensor de temperatura e umidade
DHT11**



Fonte: elaborado pelo autor.

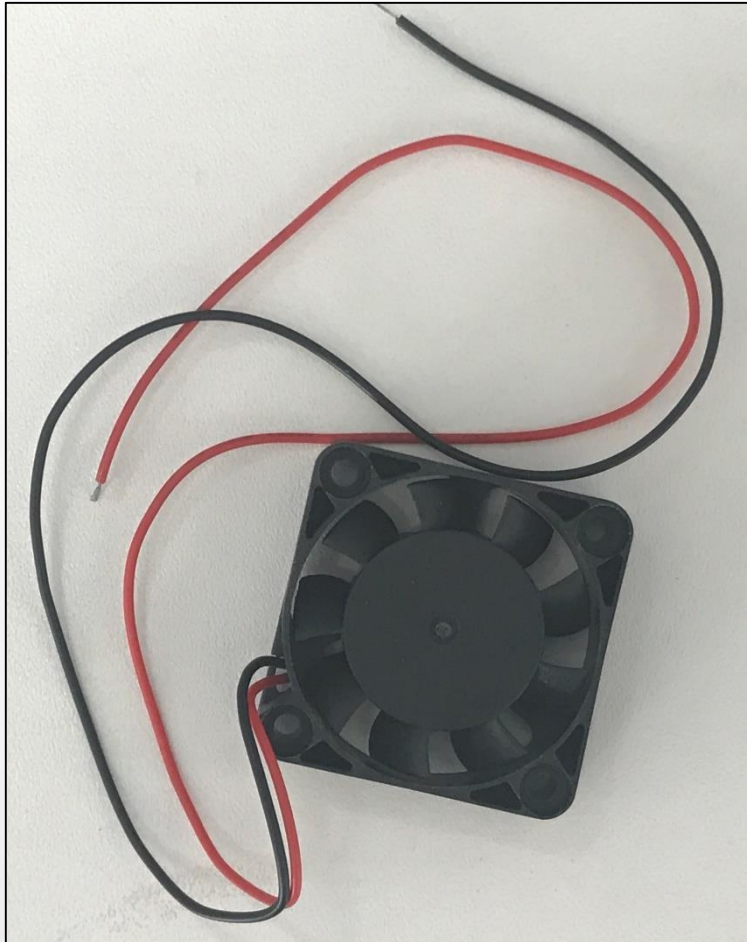
**Buzzer Ativo
Bip Contínuo**



Fonte: elaborado pelo autor.

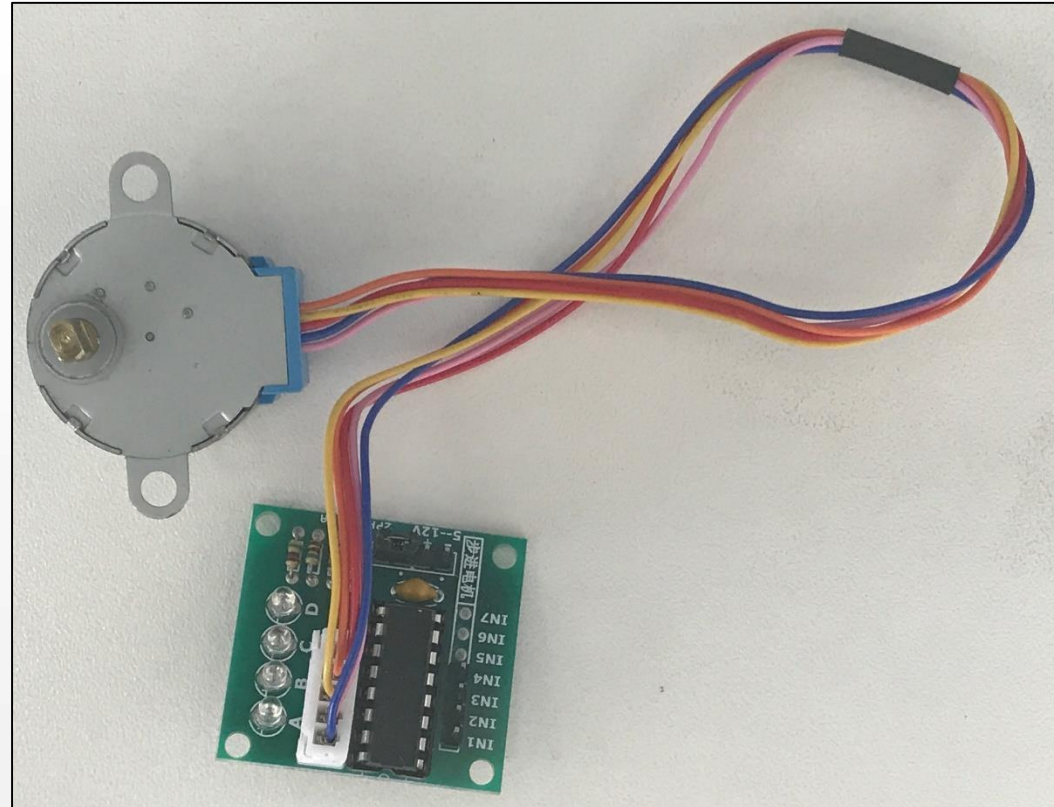
Hardware do sistema

Mini Cooler



Fonte: elaborado pelo autor.

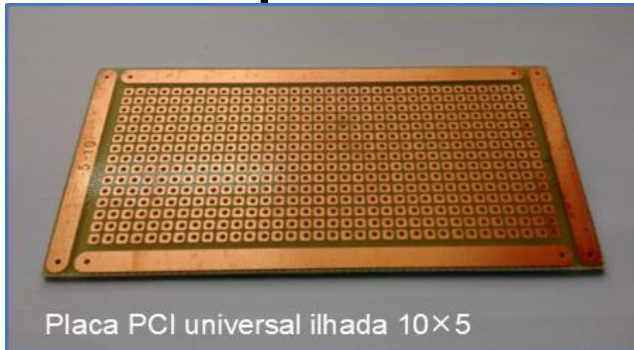
Motor de passo 28BYJ-48 conectado ao Módulo Driver Uln2003



Fonte: elaborado pelo autor.

Hardware do sistema

Os demais componentes utilizados no hardware do sistema



Placa PCI universal ilhada 10×5



Resistores de 220Ω



Fios jumpers



LEDs brancos e verdes Straw Hat (2.4V)



Fonte de Alimentação 5V

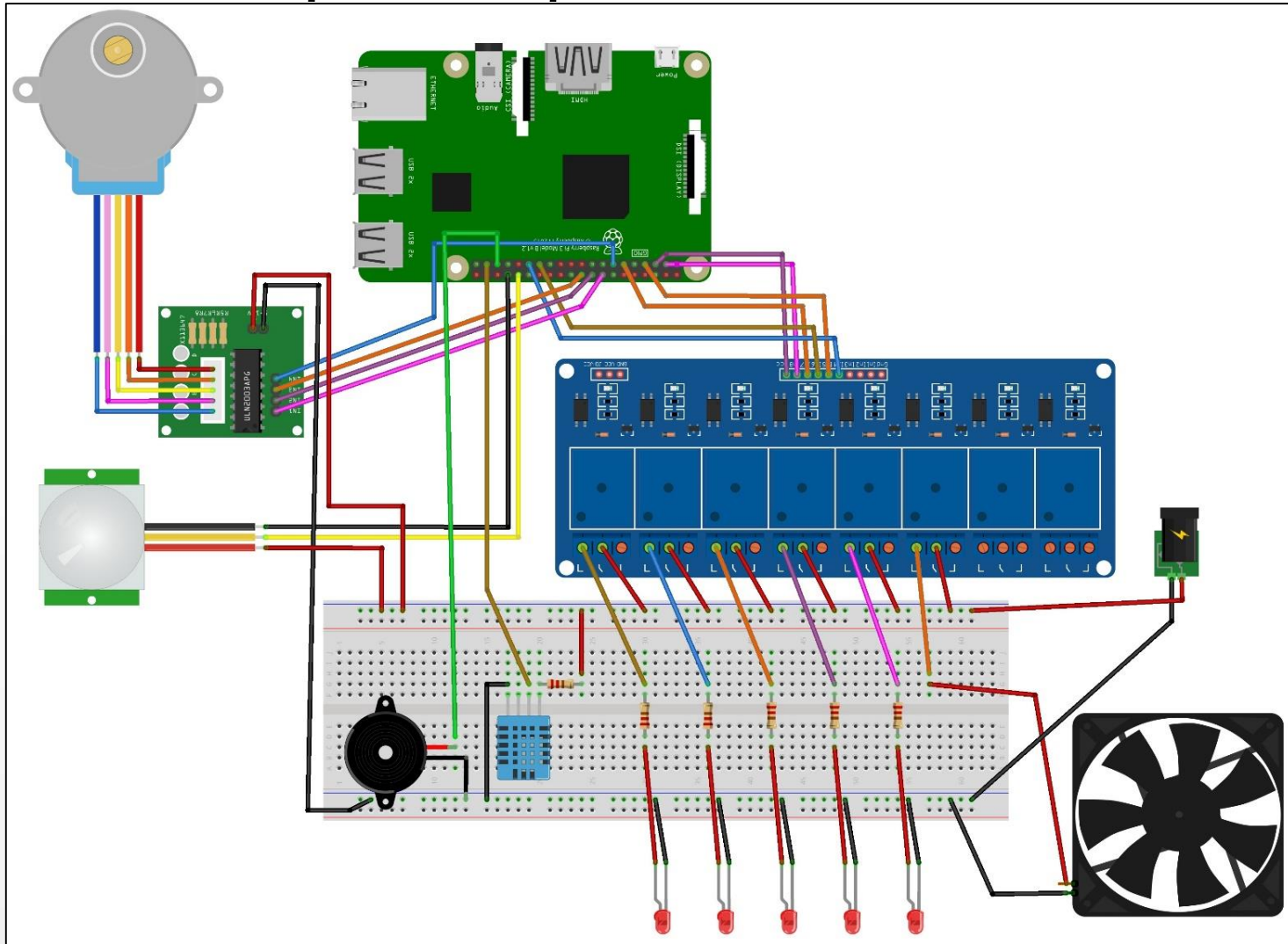
Fonte: elaborado pelo autor.



Fonte de Alimentação 12V

Hardware do sistema

Esquema completo da central de controle



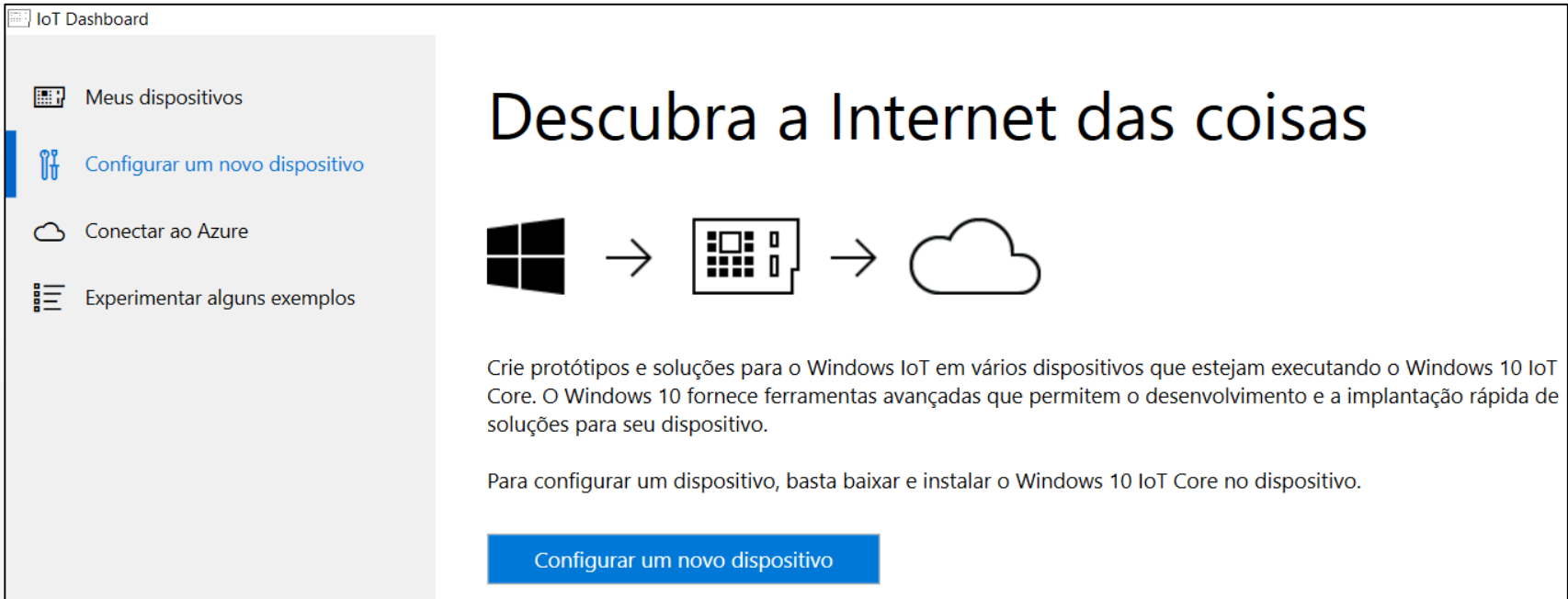
Fonte: elaborado pelo autor.

Técnicas e ferramentas

2. Desenvolvimento do Software

Desenvolvimento do Software

Instalação do Windows 10 IoT Core no Raspberry Pi 3 Model B através do Windows 10 IoT Core Dashbord



The screenshot shows the 'IoT Dashboard' interface. On the left, there is a sidebar with four menu items: 'Meus dispositivos', 'Configurar um novo dispositivo' (highlighted in blue), 'Conectar ao Azure', and 'Experimentar alguns exemplos'. The main content area features the heading 'Descubra a Internet das coisas' and a diagram showing the Windows logo, an arrow, a Raspberry Pi icon, another arrow, and a cloud icon. Below the diagram, there is a paragraph of text and a blue button labeled 'Configurar um novo dispositivo'.

IoT Dashboard

- Meus dispositivos
- Configurar um novo dispositivo**
- Conectar ao Azure
- Experimentar alguns exemplos

Descubra a Internet das coisas

Windows → Raspberry Pi → Cloud

Crie protótipos e soluções para o Windows IoT em vários dispositivos que estejam executando o Windows 10 IoT Core. O Windows 10 fornece ferramentas avançadas que permitem o desenvolvimento e a implantação rápida de soluções para seu dispositivo.

Para configurar um dispositivo, basta baixar e instalar o Windows 10 IoT Core no dispositivo.

[Configurar um novo dispositivo](#)

Fonte: elaborado pelo autor.

Software embarcado



- **Configuração para habilitar o computador para o modo desenvolvedor;**
- **Instalação da extensões Windows IoT Project Templates & Windows IoT Extensions for the UWP.**

Inicialização do servidor, conexão, processamento e retorno do resultado

```
using Windows.Networking.Sockets;
using Windows.Storage.Streams;
//...
private async void inicializa(StreamSocketListener sender,
                               StreamSocketListenerConnectionReceivedEventArgs args) {

    StringBuilder requisicao = new StringBuilder();

    using (IInputStream input = args.Socket.InputStream) {
        byte[] data = new byte[TAMANHO_BUFFER];
        IBuffer buffer = data.AsBuffer();
        uint leituraDado = TAMANHO_BUFFER;

        while (leituraDado == TAMANHO_BUFFER) {
            await input.ReadAsync(buffer, TAMANHO_BUFFER, InputStreamOptions.Partial);
            requisicao.Append(Encoding.UTF8.GetString(data, 0, data.Length));
            requisicao.Append(Encoding.UTF8.GetString(data, 0, data.Length));
            leituraDado = buffer.Length;
        }
    }
    string informacaoRecebida = requisicao.ToString().Substring(0, 7);
    string informacaoResposta = "";

    switch (informacaoRecebida) {

        case "GPIO02V":
            hajaLuz(pinoGpio02, true);
            informacaoResposta = $"Luz da Sala ligada!";
            break;

            //...

    }

    using (IOOutputStream output = args.Socket.OutputStream)

    using (Stream resposta = output.AsStreamForWrite()) {
        var header = Encoding.UTF8.GetBytes($"\\n{informacaoResposta}\\n");
        await resposta.WriteAsync(header, 0, header.Length);
    }
}
```

Captura da detecção de movimentos

```
using Windows.Devices.Gpio;
//...

private void inicializaGPIO() {
    var gpio = GpioController.GetDefault();

    //Sensor de movimento
    pinoGpio12 = gpio.OpenPin(12);
    pinoGpio12.SetDriveMode(GpioPinDriveMode.Input);
    alarmeAtivado = false;
    pinoGpio12.ValueChanged += detectaMovimento;

    //...
}

private async void detectaMovimento(GpioPin sender,
                                     GpioPinValueChangedEventArgs args) {
    var detectou = args.Edge == GpioPinEdge.FallingEdge;
    if (alarmeAtivado) {
        if (detectou){
            qtdAlarme++;

            agora = DateTime.Now;
            mukolo = agora.ToString("dddd", cult);

            enviaEmail();
            disparaAlarme();
            addHistorico();
        }
    }
}
```


Leitura da temperatura e umidade

```
using Sensors.Dht;
//...

private IDht dht = null;
private String resultadoDHT;

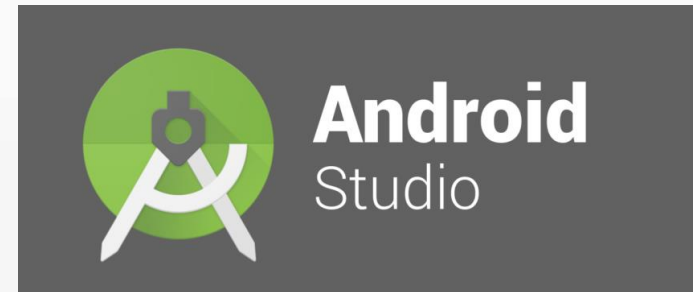
//DHT
private async Task medeDHT() {
    DhtReading reading = new DhtReading();

    reading = await dht.GetReadingAsync().AsTask();

    if (reading.IsValid) {
        this.Temperatura = Convert.ToSingle(reading.Temperature);
        this.Humidade = Convert.ToSingle(reading.Humidity);

        this.resultadoDHT = this.Temperatura + "-" + this.Humidade;
    }
}
```

Aplicativo móvel de gerenciamento



Resultados e Discussões

- Foi implementado o protótipo de um sistema de automação residencial utilizando o Raspberry Pi 3 Model B com o Windows 10 IoT Core;
- Os testes exaustivos do sistema comprovaram o bom funcionamento de todos os artefatos;
- Não foi identificado nenhuma falha de comunicação entre a central de controle e os sensores e atuadores;
- A execução das tarefas ocorre em 20 segundos no máximo;
- Todos os requisitos funcionais foram plenamente atendidos.

Conclusões

A utilização do Raspberry Pi 3 Model B com o Windows 10 IoT Core é indubitavelmente viável e pertinente.

Pode ser concluído que o trabalho atingiu os objetivos propostos.

Extensões

- a) Permitir o controle utilizando a comunicação das aplicações em redes diferentes;**
- b) Implementar o sistema de gerenciamento multiusuário;**
- c) Integrar o monitoramento por câmera;**
- d) Integrar o sensor de temperatura e umidade DHT22;**
- e) Implementar um sistema de interfone de tal forma que ao tocar o interfone o aplicativo gerenciador receba uma chamada e permita a comunicação via VoIP.**

Operacionalidade da Implementação