

UNIOESTE UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ  
CENTRO DE ENGENHARIAS E CIÊNCIAS EXATAS  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

# **Desenvolvimento de um sistema multiusuário para edifícios inteligentes**

Victor Hugo de Almeida Alicino

FOZ DO IGUAÇU

2024

Victor Hugo de Almeida Alicino

## **Desenvolvimento de um sistema multiusuário para edifícios inteligentes**

Monografia submetida à Universidade Estadual do Oeste do Paraná, Curso de Ciência da Computação Campus de Foz do Iguaçu, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Dr. Antonio Marcos Massao Hachisuca

FOZ DO IGUAÇU

2024

Victor Hugo de Almeida Alicino

## **Desenvolvimento de um sistema multiusuário para edifícios inteligentes**

---

**Dr. Antonio Marcos Massao  
Hachisuca**  
Orientador(a)

---

**Titulação Nome 1º membro da banca**  
Membro

---

**Titulação Nome 2º membro da banca**  
Membro

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Objetivos	2
1.1.1	Objetivo Geral	2
1.1.2	Objetivos Específicos	2
1.2	Estrutura do Trabalho	2
<b>2</b>	<b>Revisão Bibliográfica</b>	<b>4</b>
2.1	Edifício Inteligente	4
2.1.1	Intelligent Building	4
2.1.2	Smart Building	5
2.1.3	Sistema de Gerenciamento de Edifício	6
2.1.4	Sistema de Controle e Automação de Edifício	7
2.2	Internet das Coisas	9
2.3	Tecnologias	12
2.3.1	Tecnologias utilizadas em automação de edifícios	12
2.3.1.1	BACnet	12
2.3.1.2	KNX	12
2.3.1.3	ZigBee	13
2.3.2	Tecnologias utilizadas no desenvolvimento deste trabalho	13
2.3.2.1	Tasmota	13
2.3.2.2	MQTT	13
<b>3</b>	<b>Sistema Opus</b>	<b>15</b>
3.1	Breve explicação sobre o nome	15
3.1.1	Servidor Local: <b>Opus</b>	15
3.1.2	Aplicação Móvel: <b>Conductor</b>	15
3.1.3	Servidor Remoto: <b>Maestro</b>	15
3.2	Arquitetura	16
3.2.1	Camada Física	16
3.2.2	Dispositivos Controlados	16
3.2.3	Interfaces Gráficas de Usuário	16
3.3	Rede Local	17
3.3.1	Banco de Dados	17
3.3.2	Opus	17
3.3.3	Drivers	17

3.3.4	Tasmota . . . . .	17
3.4	Internet . . . . .	17
3.4.1	Banco de Dados . . . . .	18
3.4.2	Autenticação do Google . . . . .	18
3.4.3	Maestro . . . . .	18

# 1 Introdução

As cidades inteligentes visam melhorar a qualidade de vida dos cidadãos e promover um desenvolvimento econômico e social sustentável por meio do uso de tecnologias da informação e comunicação (TIC) [??]. Nesse contexto, os edifícios inteligentes desempenham um papel fundamental. Indivíduos passam grande parte da sua vida dentro de edifícios, crescem, estudam e se desenvolvem neles [??] [??]. Dada sua importância no cotidiano urbano, a modernização dos edifícios torna-se essencial para a implementação do conceito de cidades inteligentes. Assim, a integração de novas tecnologias pode otimizar a experiência dos ocupantes e contribuir para a sustentabilidade dessas construções.

Para que um edifício seja classificado como inteligente, é necessária a incorporação de dispositivos capazes de monitorar e controlar o ambiente. Esses dispositivos incluem sensores, responsáveis por captar os dados do ambiente em que se encontram; atuadores que realizam as ações físicas baseadas nas entradas que recebem; e controladores que coordenam a comunicação entre sensores e atuadores para garantir uma automação eficiente [??]. A implementação desses componentes possibilita desde o gerenciamento de iluminação e climatização até sistemas avançados de segurança e eficiência energética.

Nos últimos anos, dispositivos com essas características têm se tornado cada vez mais populares no meio da Internet das Coisas (do inglês, IoT). Com a significativa redução de custo dos sensores nas últimas duas décadas [??], dispositivos IoT destinados a Casas Inteligentes ganharam ampla aceitação no mercado, devido ao seu custo acessível e facilidade na instalação, com isso, estima-se que ao final da década de 2020, dos mais de 30 bilhões de dispositivos IoT conectados no mundo, 60% serão de consumidores finais[??].

Diante da crescente adoção de dispositivos IoT em Casas Inteligentes, surge a questão de por que esses dispositivos ainda não são amplamente utilizados em edifícios coletivos, como escritórios e coworking, para torná-los inteligentes. Os sistemas de automação tradicionais, projetados para grandes edifícios, costumam ser caros e complexos, enquanto os dispositivos IoT voltados ao consumidor final são acessíveis e fáceis de instalar. No entanto, esses dispositivos foram concebidos para um uso individual ou residencial, assumindo que os ocupantes do espaço são previamente conhecidos. Em edifícios coletivos, essa abordagem se torna inviável, pois a rotatividade de usuários exige um modelo flexível e adaptável de controle e segurança. Assim, há um desafio técnico e econômico a ser superado para que dispositivos IoT de Casas Inteligentes possam ser utilizados nesses ambientes de forma segura e eficiente.

Apesar dessa ampla adoção no mercado residencial, a aplicação desses dispositivos em edifícios coletivos ainda enfrenta desafios específicos. Considerando que dispositivos

IoT de Casas Inteligentes são mais acessíveis e fáceis de implementar que sistemas tradicionais de automação predial, mas não são viáveis em edifícios coletivos devido à necessidade de múltiplos usuários compartilharem o mesmo ambiente, como é possível adaptá-los para o uso em edifícios coletivos, garantindo controle adequado para múltiplos ocupantes?

Dado que dispositivos IoT de Casas Inteligentes são mais acessíveis e fáceis de implementar do que os sistemas tradicionais de automação predial, como é possível adaptá-los para o uso em edifícios coletivos, garantindo segurança e controle adequado para múltiplos ocupantes?

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Este trabalho visa desenvolver um sistema básico que demonstre a viabilidade de múltiplos usuários operarem simultaneamente um sistema de edifício inteligente.

### 1.1.2 Objetivos Específicos

Para alcançar o objetivo do trabalho, é proposto uma aplicação simplificada que permita entender a viabilidade do uso de um sistema de edifício inteligente por diversos usuários, focando no controle das unidades de climatização.

Dentre os principais objetivos específicos destacam-se:

- Realizar uma pesquisa bibliográfica sobre edifícios inteligentes e suas demandas;
- Estudar sobre as aplicações da internet das coisas em um edifício inteligente;
- Definir requisitos para um sistema de gerenciamento de um edifício inteligente;
- Propor e implementar um sistema de gerenciamento para edifícios inteligentes;

## 1.2 Estrutura do Trabalho

Este trabalho está organizado com a seguinte estrutura:

**Capítulo 2:** Revisão Bibliográfica, onde são apresentados...

**Capítulo 3:** Arquitetura do Sistema...

**Capítulo 4:** Dispositivo IoT...

**Capítulo 5:** Servidor Local...

**Capítulo 6:** Servidor Remoto...

**Capítulo 7:** Interface de Usuário...

**Capítulo 8:** Conclusão...



## 2 Revisão Bibliográfica

### 2.1 Edifício Inteligente

O conceito de edifícios com algum tipo de autonomia humana é chamado de edifício inteligente, esse conceito surge nos Estados Unidos em meados da década de 80 junto com os sistemas de automação de segurança e iluminação para edifícios [??]. Esses edifícios chamados de edifícios inteligentes têm se tornado populares nos últimos anos, mas os limites e requisitos do que um edifício precisa ter para ser considerado inteligente é algo nebuloso. O conceito de edifício inteligente em português origina de dois conceitos pouco distintos na língua inglesa, *Intelligent Buildings* e *Smart Buildings*, “Intelligent” segundo o *Oxford Learner’s Dictionaries* é aquele que é bom de aprendizado [??], enquanto “Smart” é aquele que é inteligente [??], o significado de ambas as palavras não diferem muito entre si, assim como os conceitos de *Intelligent Buildings* e *Smart Buildings*, ambos carregam as mesmas primícias, de um edifício com algum nível de automação, minimizando a interação humana [??].

#### 2.1.1 Intelligent Building

*Intelligent Buildings* vem sendo pesquisados e desenvolvidos há pelo menos três décadas [??] e existem no mínimo 30 definições diferentes para esse conceito [??], em 1990, Powell traz uma definição de edifício inteligente (Intelligent Building) comentando o que Stubbings diz em 1988,

At present the term ‘intelligent building’ is normally taken to mean ‘a building which totally controls its own environment’ [??]. This seems to imply that it is the technical control of heating and air conditioning, lighting, security, fire protection, telecommunication and data services, lifts and other similar building operations that is important a control typically given over to a management computer system. Such a definition for a conventionally intelligent building does not suggest user interaction at all. [??]

Stubbings diz que um edifício inteligente é aquele edifício capaz de controlar seu próprio ambiente e Powell expande essa ideia, definindo que um edifício inteligente é aquele capaz de controlar seus sistemas de Aquecimento, Ventilação e Ar-Condicionado (AVAC), iluminação, segurança, combate a incêndio etc. através de um sistema de gerenciamento de edifício (Building Management System). Clements-Croome em 2009 define um edifício inteligente como aquele que é responsivo aos requisitos dos seus ocupantes, organizações

e sociedade, sendo sustentável no consumo de água e energia, gerando pouca poluição e sendo funcional de acordo com as necessidades do usuário [??] e Brooks sugere que *Intelligent Building* e seu sistema de gestão (Building Management System) são essencialmente a mesma coisa [??], um sistema de controle que abrange todo o edifício, que conecta, controla e monitora a planta fixa e os equipamentos da instalação [??].

### 2.1.2 Smart Building

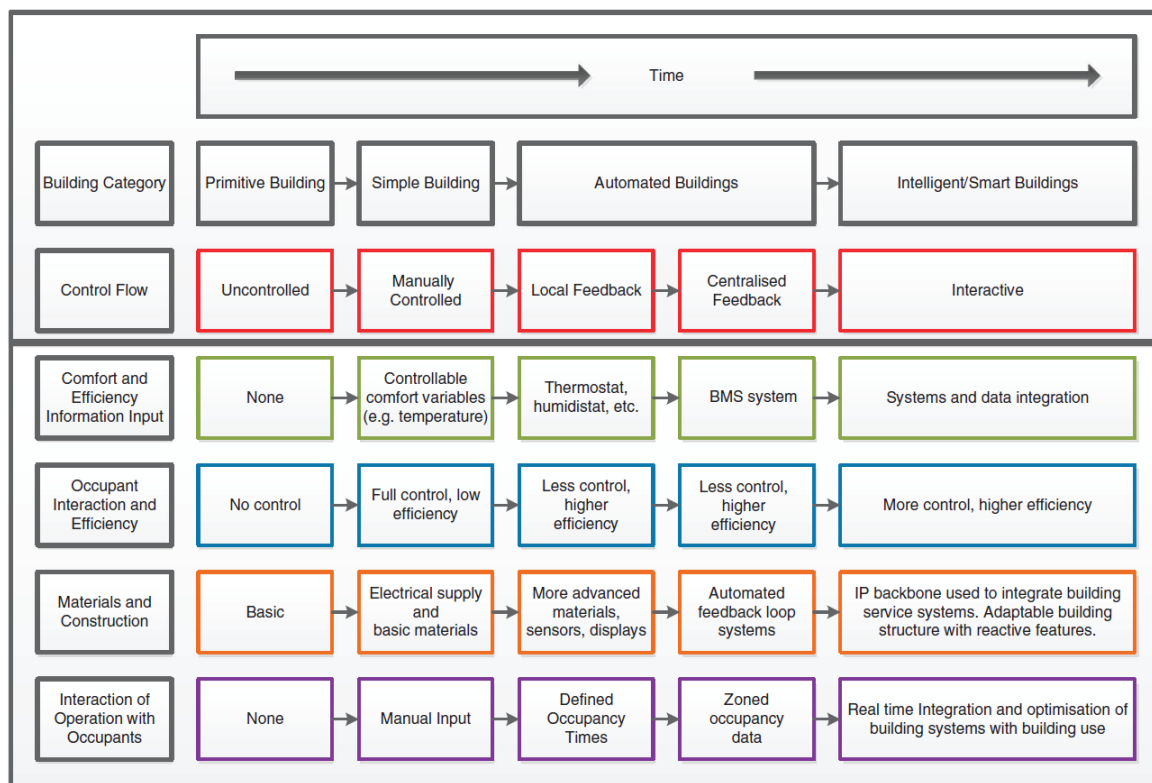


Figura 1 – Progressão da automação em edifícios

Fonte: [??]

Na figura 1, Buckman, Mayfield, Beck mostram uma linha do tempo, partindo de um edifício primitivo com nenhum controle do seu ambiente até um edifício inteligente. Segundo sua pesquisa, sugerem que existem três principais pontos que desenvolvem um edifício até o estado de edifício inteligente, são eles:

1. Longevidade;
2. Energia e Eficiência; e
3. Conforto e Satisfação.

Para atingir o estado de edifício inteligente nos três pontos, existem quatro aspectos que variam o nível do edifício de primitivo a inteligente, são eles:

1. Inteligência, a forma como são coletadas informações das operações do edifício e sua resposta;
2. Controle, a interação entre ocupantes e o edifício;
3. Materiais e Construção, a forma física do edifício; e
4. Empresa, a forma como são coletadas informações e usadas para melhorar a performance dos ocupantes.

A evolução desses métodos em um edifício partindo do primitivo ao inteligente é mostrada na figura 1. De acordo com a pesquisa, Buckman, Mayfield, Beck sugerem que em um *Smart Building*, os quatro aspectos que variam o nível do edifício são desenvolvidos lado a lado, usando a informação de um na operação do outro, diferindo de um *Intelligent Building* que desenvolve a “Inteligência” citada acima de forma independente dos outros três aspectos. *Smart Buildings* usam tanto o controle humano, quanto a automação para atingir os quatro aspectos apresentados acima. O aspecto de “Controle”, o mais importante para esse trabalho, deve apresentar informações do aspecto de “Inteligência” para os ocupantes do edifício, para que os mesmos possam se adaptar ao edifício assim como o edifício se adapta a eles[??].

Sensores inteligentes, que podem ser adicionados a qualquer momento da vida de um edifício [??], frutos da internet das coisas (IoT, do inglês Internet of Things) permitem uma rápida implementação de instalações para edifícios inteligentes, como gerenciamento do sistema de AVAC, sistemas de segurança, monitoramento por câmeras, alertas para eventos como incêndio, vazamento de gás e monitoramento da integridade estrutural do edifício [??]. A adoção de IoT promove a conectividade entre sensores, dispositivos e sistemas do edifício a nuvem, tal conexão promove o uso de aplicações que usaram os dados coletados [??].

### 2.1.3 Sistema de Gerenciamento de Edifício

Sistemas de Gerenciamento de Edifício, ou Building Management Systems (BMS) em inglês são controladores inteligentes baseados em microprocessadores instalados na rede para monitorar e controlar os aspectos técnicos de um edifício e seus serviços [??]. Um BMS pode controlar componentes com protocolos de mais baixo nível, como BACnet, Modbus e etc. [??] os subsistemas do BMS ligam as funcionalidades individuais desses equipamentos para que eles possam operar como um único sistema [??].

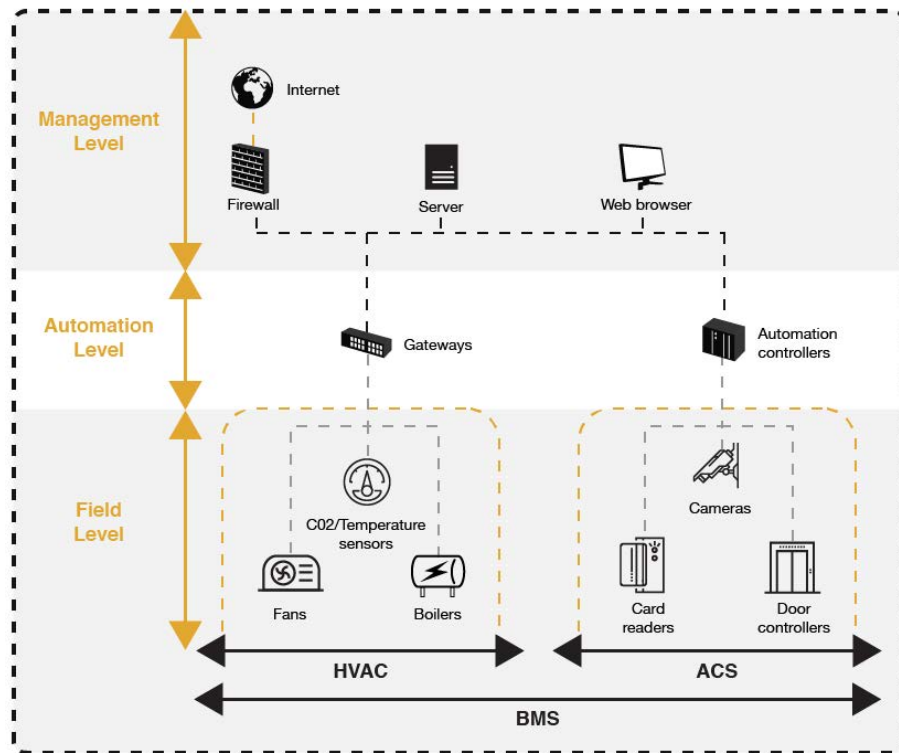


Figura 2 – Relação entre os componentes do BMS e sistemas

Fonte: [?]

A figura 2 mostra um exemplo da relação entre os componentes de um BMS.

#### 2.1.4 Sistema de Controle e Automação de Edifício

Quando um Sistema de Gerenciamento de Edifício passa a satisfazer os requisitos da ISO 16484 [?], tais como suporte a BACnet ele passa a ser um Sistema de Controle e Automação de Edifício (Building Automation Control System, BACS) [?]. Os conceitos de BMS e BACS possuem mais similaridades do que diferenças sendo difícil encontrar na literatura materiais que os diferenciem bem, sendo assim, para este trabalho, Sistemas de Gerenciamento de Edifícios (BMS) e Sistemas de Controle e Automação de Edifícios (BACS) serão considerados sinônimos.

Assim como o Sistema de Gerenciamento de Edifício, o Sistema de Controle e Automação de Edifício também é responsável por controlar e monitorar os aspectos técnicos de um edifício e seus serviços, o BACS também é dividido em três camadas, como representado na figura 2, sendo elas:

- Camada de Campo (*Field Layer*);
- Camada de Automação (*Automation Layer*); e
- Cada de Gerenciamento (*Management Layer*).

A camada de campo é a camada mais baixa, sendo nela encontrados os sensores e atuadores que fazem a interação com o ambiente. A camada de automação é onde os dados são processados, loops de controle são executados e alarmes ativados. Por último, a camada de gerenciamento é a responsável por apresentar os dados do sistema, criar logs de acontecimentos e intermediar o controle do usuário com o sistema. Sistemas mais modernos tendem a separar a lógica da interface gráfica para o usuário com o objetivo de criar acessos mais flexíveis aos BACS [??].

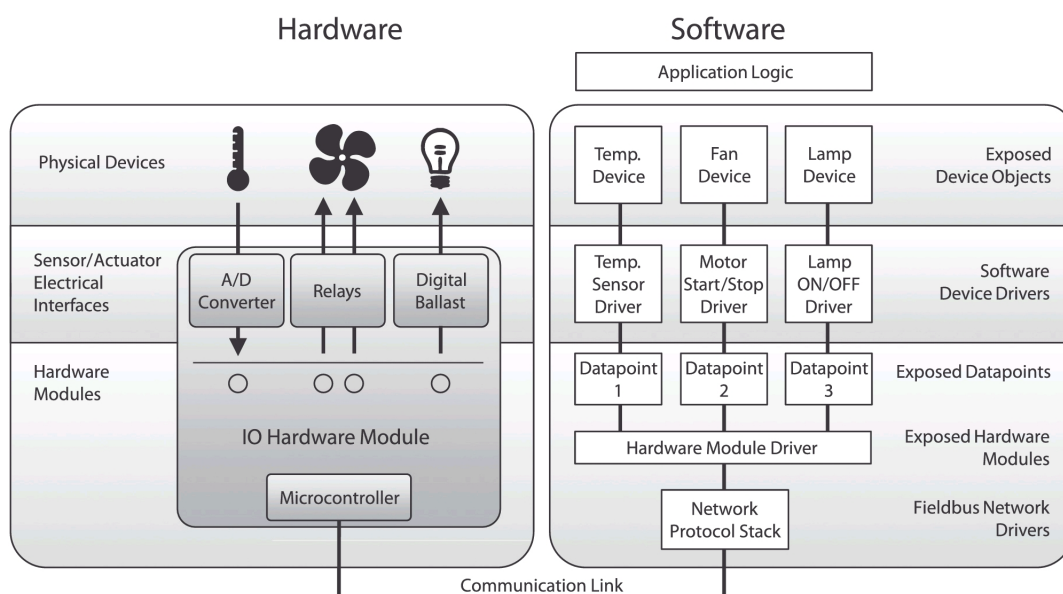


Figura 3 – Pilhas de Hardware e Software envolvidos no Sistema de Controle e Automação de Edifício

Fonte: [??]

O caminho da informação em um BACS tem a forma representada pela figura 3, sensores e atuadores interagem com dispositivos físicos através de módulos em hardware, controlados por um microcontrolador responsável por enviar os dados até o BACS através de uma conexão de comunicação (também chamada de barramento de campo ou *fieldbus* em inglês) através de um dos protocolos de baixo nível já citados, como BACnet, KNX, LonWorks, Modbus, ZigBee ou EnOcean. Os dados são recebidos por módulos de hardware que expõem esses dados para o software através de *datapoints* (ou também chamados de *endpoints*, *tags* ou *points*) que interagem com o software para realizar a ação necessária [??].

## 2.2 Internet das Coisas

Enquanto a internet foi desenvolvida com dados criados por pessoas, a Internet das Coisas é sobre os dados criados por objetos [??]. Madakam, Ramaswamy e Tripathi definem Internet das Coisas como uma rede de objetos inteligente aberta que tem a capacidade de se auto-organizar, compartilhar informações, dados e recursos [??].

Internet das Coisas é muitas vezes referido pela sigla **IoT** que vem do seu nome em inglês *Internet of Things*. No nome dessa tecnologia temos as duas partes que desempenham os principais papéis: *Internet* e *Coisas*. *Internet* aqui, se refere a mesma internet usadas por bilhões de pessoas ao redor do mundo, já *Coisas*, se refere aos dispositivos com capacidades computacionais atrelados a sensores ou atuadores conectados à rede, o que permite a eles trocarem e consumirem informações.

Em uma rede IoT, dispositivos (muitas vezes chamados de *inteligentes*) tem a habilidade de se comunicarem para monitorar o ambiente em que estão, ou alterar este ambiente. Essas ações podem ser configuradas previamente pelo usuário ou serem definidas na hora com a possibilidade de ser acionadas fora da rede local onde esses dispositivos estão conectados através da internet. Como por exemplo, verificar a temperatura de uma sala sem ter chego nela e solicitar ao sistema que acione as unidades AVAC para que a sala esteja na temperatura desejada ao chegar.

Apesar de ocorrer leves diferenças nos modelos de arquitetura, geralmente um sistema IoT é formado por três camadas, sendo elas:

- Camada Física: responsável por perceber o ambiente físico com sensores, atuadores ou qualquer tipo de interface para perceber e modificar o ambiente físico;
- Camada de Rede: camada de transmissão dos dados através das inúmeras formas de conexões disponíveis como redes sem fio, redes móveis e a internet, a fim de fornecer os dados da Camada Física para a Camada de Aplicação;
- Camada de Aplicação, que oferece os serviços chamados *inteligentes* para os usuários finais.

[??].

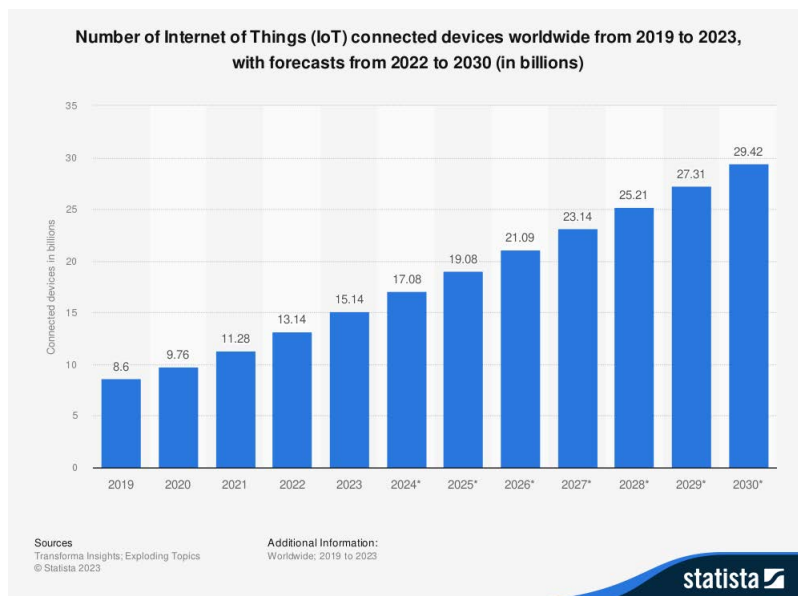
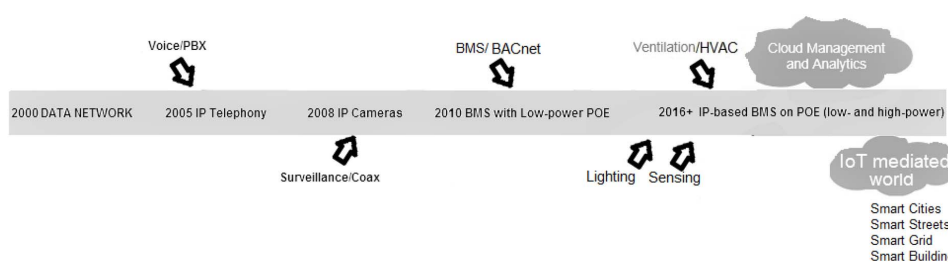


Figura 4 – Números de dispositivos IoT conectados no mundo de 2019 à 2023

Fonte: ??]

Segundo a empresa especializada em coleta de dados, Statista, a quantidade de dispositivos IoT conectados no mundo passa de 15 bilhões em 2023 com projeção para quase o dobro deste número até o fim da década [??] como mostrado na figura 4.

Isso se deve muito ao fato de que sensores e microcontroladores ficaram mais baratos, o preço médio dos sensores caiu 200% entre 2004 e 2018 [??] o que significa mais produtos finais voltado para IoT já que ficou possível embarcar sensores em itens do dia a dia com custos acessíveis.



Note: BACnet is an ASHRAE, ANSI, and ISO 16484-5 standard communications protocol for building automation and control (BACnet was subsumed in ASHRAE/ANSI Standard 135 in 1995, and in ISO 16484-5 in 2003.) The BACnet protocol defines several services that are used to communicate between control devices typically utilized in building (including HVAC, lighting control, access control, and fire detection systems). It specifies a number of network, data link, and physical layer protocols, including but not limited to standards such as IP/Ethernet.

Figura 5 – Convergência da tecnologia de sistemas inteligentes para edifícios nos últimos anos

Fonte: ??]

Alguns anos atrás, mesmo em edifícios com BACS, não era incomum encontrar diferentes protocolos, redes e cabos, o que gerava ineficiência tanto na implantação como nos sistemas de gerenciamento. Dessa forma algumas soluções migraram para o uso de cabo e conjunto de protocolos comuns entre as aplicações, como o cabo de par trançado

categoria 6 e o protocolo TCP/IP. Com a convergência para protocolos como TCP/IP facilitou a entrada do IoT em BACS, como mostrado na figura 5 [??].

Já é possível encontrar equipamentos preparados para uso em BACS com suporte a protocolos IoT, como o PFC200 da WAGO que suporta o protocolo MQTT (citado na seção 2.3.2.2).



Figura 6 – Controlador PFC200

Fonte: [??]



## 2.3 Tecnologias

### 2.3.1 Tecnologias utilizadas em automação de edifícios

Nesta sessão serão apresentados algumas tecnologias utilizadas na automação de edifícios

#### 2.3.1.1 BACnet

Protocolo de Rede de Automação e Controle de Edifícios (Building Automation and Control Network Protocol) ou BACnet é um protocolo mantido e desenvolvido pela ASHRAE com o objetivo de atender as necessidades de comunicações dos sistemas de controle e automação dos edifícios com seus sensores e atuadores. O protocolo fornece meios para que equipamentos troquem informações não importando o serviço que eles realizem [??]. A forma com que o BACnet funciona, permite que usuários não fiquem presos a sistemas proprietários, criando um padrão de comunicação entre os dispositivos como sistemas de AVAC, alarmes de incêndio e etc. uma das suas grandes vantagens é a habilidade de se adaptar a novas tecnologias de rede [??]. o BACnet define um modelo de informação, criando objetos para os dispositivos conectados, esses objetos representam seus serviços, assim como suas entradas e saídas. Esses objetos chamados de *device object* definem as propriedades do dispositivo como: nome do modelo, fabricante, status e a lista de outros objetos BACnet associados ao dispositivo [??].

#### 2.3.1.2 KNX

KNX ou Konnex é um padrão aberto desenvolvido para ser utilizado em automação de edifícios ou residências [??], da mesma forma que o BACnet, o KNX integra sensores, atuadores e controladores dentro de uma rede, os dispositivos são controlados através de um barramento e são chamados de *Bus Access Unit* ou BAU, sendo possível controlar mais de 65 mil BAUs. No KNX, cada fabricante pode implementar sua própria especificação nos dispositivos. BAUs podem ser separadas em grupos menores chamados de *areas* que podem ser separadas em *lines* de dispositivos. As mensagens trocadas pelos dispositivos KNX pode ser transmitida através de cabos de par trançado, cabos de energia, frequências de rádio ou internet, sendo a forma mais comum o KNX.TP que usa os cabos de par trançado [??]. Um ponto chave do KNX é seu método de comunicação, pouco similar ao MQTT, o KNX utiliza o padrão *observador* na hora de trocar informações, neste padrão múltiplos dispositivos “observadores” serão notificados através de uma única mensagem em *multicast* de quando os dados em uma fonte sofrerem alterações, dessa forma fica mais fácil a criação de relacionamentos um para muitos (1:N). [??].

### 2.3.1.3 ZigBee

ZigBee é um dos mais populares padrões de redes *mesh* sem fio. O protocolo é aberto e baseado em pacotes, projetado para ser de fácil uso, baixo consumo e seguro [??]. O modelo ZigBee é principalmente definido em três camadas.

- *Application Support Sublayer*: Responsável por vincular *endpoints*, transmitir mensagens entre os dispositivos e o gerenciamento de grupos.
- *ZigBee Device Object*: Responsável pelo gerenciamento de dispositivos; definir o modo de operação do dispositivo; descobrir novos dispositivos e quais serviços de aplicação ele provê; lidar com os pedidos de vinculamento de outros dispositivos.
- *Application Framework*: Abriga as aplicações dos dispositivos.

[??]

## 2.3.2 Tecnologias utilizadas no desenvolvimento deste trabalho

### 2.3.2.1 Tasmota

Tasmota é um software de código aberto que atua como firmware para SoC's (System-on-a-Chip) ESP32 da Espressif e possibilita controle e comunicação personalizado com alguns dispositivos comerciais que são baseados nesse SoC, como interruptores inteligentes, lâmpadas inteligentes ou controles infravermelho inteligentes. A comunicação com o firmware é realizada através do protocolo MQTT, permitindo integrar o hardware comercial a qualquer sistema com suporte ao protocolo [??].

### 2.3.2.2 MQTT

MQTT ou Message Queuing Telemetry Transport [??] é um protocolo de transporte para mensagens no formato *publicar/assinar*, ele é simples, leve, aberto e projetado para ser fácil de implementar, seu foco de uso são ambientes onde recursos são geridos com cuidado, como comunicação *Machine to Machine* (M2M) e Internet das Coisas [??]. O foco principal do projeto desse protocolo é minimizar o consumo de banda e recursos do dispositivo, sendo assim, ele é capaz de transmitir dados com uma banda baixa e conexões instáveis. O protocolo funciona sobre TCP/IP ou outros protocolos de rede que possam prover as mesmas capacidades do TCP/IP como conexões bi-direcionais. O MQTT oferece três qualidades de serviços (*Quality of Service*, ou QoS) na entrega de mensagens.

- QoS 0: “No máximo uma vez”, mensagens são entregues no máximo uma vez.

- QoS 1: “Pelo menos uma vez”, mensagens são garantidas de serem entregues pelo menos uma vez, pode ocorrer duplicatas.
- QoS 2: “Exatamente uma vez”, mensagens são entregues exatamente uma vez.

Uma estrutura que suporte o protocolo MQTT deve ter pelo menos três componentes,

1. *Publisher* ou Produtor, um cliente MQTT
2. *Broker*, o servidor MQTT que gerencia o recebimento e entrega das mensagens
3. *Subscriber* ou Consumidor, um cliente MQTT

No padrão *publicar/assinar*, o produtor cria uma mensagem e envia para o servidor, consumidores então leem essa mensagem. No MQTT um produtor publica mensagens nos tópicos do *broker*, consumidores interessados em receber essas mensagens se inscreverão nesse tópico e o *broker* se encarregará de entregá-las [??].

## 3 Sistema Opus

Buscando o objetivo do trabalho, foi proposto um sistema denominado de *Opus*, dividido em três partes, **Opus**, **Maestro** e **Conductor**, ele tem o objetivo principal de permitir que múltiplos usuários possam controlar dispositivos IoT.

### 3.1 Breve explicação sobre o nome

Como uma homenagem à música erudita, os componentes do sistema foram nomeados com algumas palavras recorrentes no meio musical. A seguir, uma breve descrição de cada componente e o motivo de seu nome.

#### 3.1.1 Servidor Local: **Opus**

A palavra *opus* vem do latim e quer dizer *obra* [??]. Na música erudita o termo é utilizado para identificar uma obra específica de um compositor [??], como o caso da *Op. 67* de Beethoven (Op. é a abreviação de *opus*), esta composição é conhecida por vários nomes como *Sinfonia n.º 5* ou *Sinfonia do Destino* [??], então para identificá-la é usado o *Opus Number*, ou número da obra.

O *Servidor Local* leva este nome, pois é nele que encontramos os dispositivos, quase como instrumentos em uma orquestra, os dispositivos são parte de uma obra cujo objetivo é a coordenação de um edifício.

#### 3.1.2 Aplicação Móvel: **Conductor**

A aplicação móvel leva o nome em inglês de *Conductor*, que significa *Condutor*, o condutor é o responsável por dirigir a orquestra ou banda durante a música.

#### 3.1.3 Servidor Remoto: **Maestro**

O *Servidor Remoto* leva o nome de *Maestro* sem muito motivo especial, a palavra vem do italiano e significa *mestre* [??], maestro é um título de respeito atingido por um condutor de orquestra. Apenas por uma questão hierárquica, o *Servidor Remoto* foi nomeado de *Maestro*.

## 3.2 Arquitetura

### 3.2.1 Camada Física

A arquitetura proposta é ilustrada na figura 7.

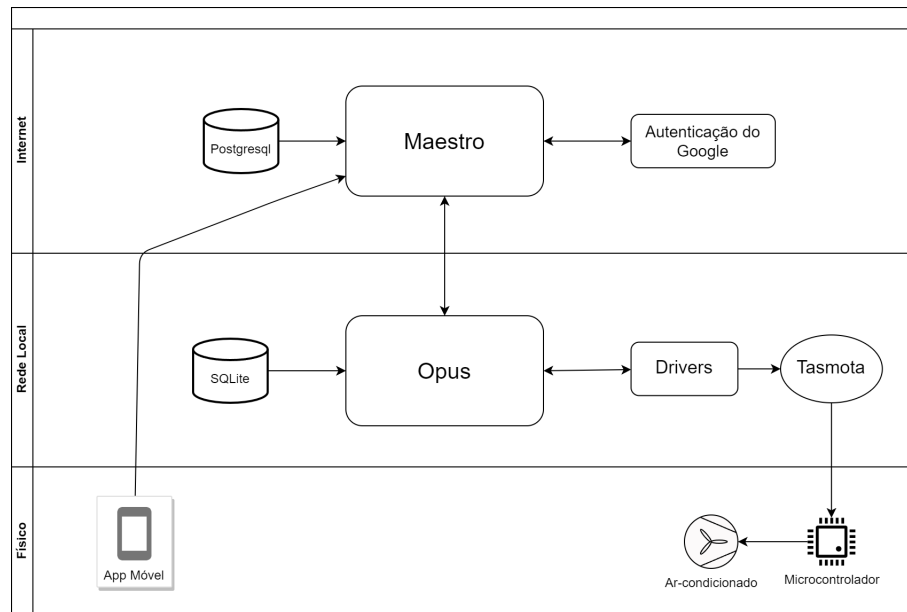


Figura 7 – Arquitetura completa

Fonte: Elaborado pelo Autor (2024)

Camada física é o nome dado por este trabalho a tudo que tem por característica principal, seu hardware, como é o caso de um microcontrolador e um smartphone.

Nesta camada estão dispostos os dispositivos controlados pelo *Opus* e as interfaces gráficas de usuário (GUI) como o aplicativo móvel e o aplicativo web.

### 3.2.2 Dispositivos Controlados

O sistema *Opus* é feito para ser possível controlar qualquer dispositivo que o desenvolvedor projetar um *driver* para ele. Esses dispositivos em sua forma física normalmente são constituídos de um microcontrolador com acesso à internet e um ou mais atuadores.

### 3.2.3 Interfaces Gráficas de Usuário

As interfaces gráficas de usuário são os meios que o usuário tem para interagir com o sistema. Nesta arquitetura temos duas interfaces gráficas de usuário: o aplicativo móvel e o aplicativo web. O aplicativo móvel é feito para servir de ponto principal de acesso para o usuário, podendo interagir com os *Dispositivos Controlados* pelo *Opus* já o aplicativo web é a forma que o administrador tem de interagir com o *Maestro*.

## 3.3 Rede Local

A rede local refere-se a conexão local dentro de um ambiente, como a rede interna de um edifício. Ela pode ou não estar conectada a internet já que a maioria dos protocolos de comunicação utilizado pelos *Dispositivos Controlados* não requerem conexão com a internet, apenas à uma rede local com DHCP.

### 3.3.1 Banco de Dados

Dentro da rede local deve estar situado o banco de dados que o *Opus* utiliza para armazenar a estrutura do edifício e os dados dos *Dispositivos Controlados*. Devido ao fato do *Opus* ser toda uma aplicação contida dentro de si mesma, sem dependências além das bibliotecas utilizadas na sua concepção e das escolhidas pelo usuário como os *Drivers*, o banco de dados escolhido foi o SQLite3, assim não é necessário que o usuário precise instalar outro software para que o *Opus* funcione.

### 3.3.2 Opus

O *Opus* é o núcleo do sistema, a aplicação propriamente dita, responsável por se comunicar com os *Dispositivos Controlados* e manter as estruturas do edifício. O *Opus* requer comunicação constante com o *Maestro* para receber comandos do usuário, logo, apesar de estar situado na Rede Local, que por sua vez não necessita de conexão com a internet, o *Opus* necessita de uma.

### 3.3.3 Drivers

Os *Drivers* são as partes do *Opus* que podem ser estendidas por outros programadores, *Drivers* são responsáveis por se comunicar com os *Dispositivos Controlados*, traduzindo os comandos do *Opus* para o protocolo do dispositivo, eles são desenvolvidos como módulos do Python, e são carregados dinamicamente pelo *Opus*.

### 3.3.4 Tasmota

Para provar a viabilidade do sistema, foi escrito um *Driver* para o *Opus*, um driver para comunicação com o firmware Tasmota 2.3.2.1.

## 3.4 Internet

Na camada de internet está o *Maestro*, o servidor central na nuvem, onde ficam armazenados os usuários e os servidores *Opus* cadastrados.

### 3.4.1 Banco de Dados

O banco de dados da camada de internet é o responsável por armazenar os usuários e os servidores *Opus* cadastrados, este banco de dados não necessariamente precisa estar na mesma máquina que o *Maestro*, sendo assim, existe a possibilidade de configurar um banco de dados externo. A escolha do banco de dados foi o PostgreSQL, por ser um banco de dados de código aberto e versátil.

### 3.4.2 Autenticação do Google

Inspirado no modelo *Zero Trust* ou *Confiança Zero* [??], o sistema não tenta implementar autenticação de usuários e sim delega essa responsabilidade para uma entidade externa que pode verificar a autenticidade do usuário, aqui foi escolhido o serviço de autenticação da empresa Google.

### 3.4.3 Maestro

O *Maestro* é o servidor central do sistema, responsável por manter os usuários e os servidores registrados, além de ser o responsável por intermediar a comunicação entre os servidores *Opus* e os usuários.