SISTEMA IOT PARA CONTROLE DE CASA INTELIGENTE

Ismael de Jesus Barbosa Silva, Thiago Moreira de Almeida Trabuco Lima

# Resumo

Este trabalho apresenta o desenvolvimento de um sistema IoT para controle residencial baseado em ESP32, com conectividade via Wi-Fi e integração com o Firebase. O sistema permite o acionamento remoto e automático de periféricos, como luzes, sensores e displays LCD, com possibilidade de expansão para controle completo da energia elétrica da casa. Futuramente, pretende-se implementar uma arquitetura modular com sensores infravermelhos e controle centralizado por aplicativo móvel.

Palavras-chave: Automação residencial, ESP32, Firebase, Casa inteligente, IoT.

# 1. Introdução

A automação residencial tem ganhado destaque por promover comodidade, segurança e economia energética. Este projeto visa transformar uma residência em uma casa inteligente através do controle remoto e autônomo de periféricos essenciais. Utilizando o microcontrolador ESP32, o sistema atual já permite o acionamento e desligamento de luzes e displays via Firebase, além de resposta automática a eventos como detecção de movimento.

# 2. Objetivos

- Desenvolver um sistema funcional de controle de periféricos domésticos com ESP32.  
- Integrar sensores e atuadores (como LEDs, relés e LCD) ao Firebase para controle remoto.  
- Estabelecer modos de operação automáticos e manuais.  
- Permitir expansão futura para controle total da energia da residência.  
- Projetar uma solução escalável com módulos distribuídos e controle central por aplicativo.  
- Incluir um sistema de infravermelho para controle de aparelhos eletrônicos comuns.  
- Validar as conexões físicas com QR Code e instalação profissional.

# 3. Materiais e Métodos

## 3.1 Hardware utilizado:

- ESP32  
- Sensor de movimento  
- Display LCD 16x2  
- Relés  
- LEDs

## Imagens ilustrativas do protótipo.

## 3.2 Software e conectividade:

- Linguagem: C++ (Arduino Framework)  
- Firebase Realtime Database para leitura e escrita de comandos  
- Comunicação via Wi-Fi  
- Modo manual e automático com alternância dinâmica  
- Serial Monitor como fallback de controle local

## 3.3 Funcionalidades principais:

- Acionamento automático de LED baseado em movimento  
- Controle manual ou remoto dos dispositivos via Firebase  
- Atualização de estado dos periféricos no banco de dados em tempo real  
- Comandos remotos lidos do Firebase (ligar/desligar luz e LCD)  
- Serial input para depuração e comandos locais

# 4. Resultados Parciais

O sistema já é capaz de:  
- Operar em modo automático, acionando luzes quando movimento é detectado.  
- Responder a comandos do Firebase para ligar/desligar periféricos.  
- Alternar entre modo automático e manual.  
- Exibir mensagens no LCD em tempo real baseadas no estado do sistema.

# 5. Perspectivas Futuras

- Criação de um gerenciador central para coordenar todos os ESP32 distribuídos pela casa, permitindo comunicação entre os dispositivos para tomada de decisões conjunta e operação em rede.  
 Exemplos de aplicação:  
 • Um único comando do aplicativo pode ser disseminado a todos os módulos;  
 • Sensores de presença em diferentes ambientes informam uns aos outros para um controle de iluminação inteligente;  
 • Se um microcontrolador falhar, outro pode assumir temporariamente sua função;  
 • Integração com sistemas externos como Alexa ou Google Home pode ser feita apenas no módulo central;  
 • Possibilidade de balanceamento de carga e shutdown seletivo baseado em consumo;  
- Implementação de módulos distribuídos em pontos estratégicos da casa.  
- Controle de carga elétrica total por cômodo, com medição e corte de energia.  
- Inclusão de sensor infravermelho para controle de TVs, AC e outros aparelhos.  
- Desenvolvimento de um aplicativo móvel com QR code para validação e gerenciamento.  
- Implementação de melhorias na modularidade dos periféricos, permitindo que cada módulo seja configurado dinamicamente através de um banco de dados.  
 Exemplos de aplicação:  
 • Um novo tipo de sensor ou atuador pode ser adicionado sem reprogramar o firmware;  
 • Parâmetros como tempo de resposta, ações padrão e vínculos com outros dispositivos serão definidos no banco de dados;  
 • Facilita manutenção, personalização e escalabilidade do sistema para diferentes casas ou cômodos.  
- Construção de maquete física para testes em escala reduzida.