

CENTRO UNIVERSITÁRIO - UNIOPET
CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PROJETOR INTELIGENTE

CAETANO MOREIRA
GUSTAVO HENRIQUE
JOEL SÉRGIO
RAFAEL PESSOA
TIAGO BARBIERI

CURITIBA / PR
2025

PROJETOR INTELIGENTE

Proposta do Projeto Integrador (PI), apresentado como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas da Faculdade de Tecnologia UniOpet.

**Professor orientador(a):
Claudio Bilyk**

**CURITIBA/PR
2025**

SUMÁRIO

1. INTRODUÇÃO.....	4
2. OBJETIVO GERAL	5
3. OBJETIVOS ESPECÍFICOS	6
4. JUSTIFICATIVA	7
5. DOCUMENTAÇÃO DE REQUISITOS.....	8
6. REQUISITOS FUNCIONAIS.....	9
7. REQUISITOS NÃO FUNCIONAIS.....	10
8. ARQUITETURA DO SOFTWARE	11
9. DIAGRAMA ELETRÔNICO	13
10. CÓDIGO DO PROJETO EM PYTHON.....	14
11. PROJETO CONCEITUAL.....	15
12. CRONOGRAMA DE EXECUÇÃO	16
13. TECNOLOGIAS UTILIZADAS.....	17
14. CONCLUSÃO E HABILIDADES DESENVOLVIDAS.....	18
15. LINK DO PROJETO NO TINKERCAD	20
16. LINK DO PROJETO HOSPEDADO NO GITHUB	21
17. REFERÊNCIAS	22

1. INTRODUÇÃO

A educação é um dos aspectos mais importantes para o desenvolvimento de uma sociedade. Com o avanço da tecnologia, novas ferramentas têm sido incorporadas ao ambiente educacional, visando melhorar a qualidade do ensino e facilitar o trabalho dos professores. No entanto, muitos desses recursos ainda apresentam limitações que dificultam sua utilização em sala de aula.

Um dos maiores desafios enfrentados pelos educadores é a necessidade de gerenciar diversos dispositivos tecnológicos durante as aulas, o que pode consumir tempo e desviar a atenção do conteúdo a ser ensinado.

Diante desse cenário, este projeto propõe o desenvolvimento de um projetor inteligente baseado em Internet das Coisas (IoT), que funcione através de comandos de voz. A implementação desse dispositivo visa proporcionar uma solução prática para os professores, permitindo que eles controlem o projetor sem a necessidade de interação manual.

2. OBJETIVO GERAL

Desenvolver um projetor multimídia inteligente utilizando fundamentos de Internet das Coisas (IoT) que funcione através de comandos de voz, utilizando a plataforma Arduino. O dispositivo permitirá o controle do projetor de forma eficiente, sem a necessidade de interação manual.

3. OBJETIVOS ESPECÍFICOS

1. **Integrar a plataforma Arduino:** Utilizar a plataforma Arduino para desenvolver o hardware necessário para o funcionamento do projetor inteligente.
2. **Desenvolver sistema de comandos de voz:** Implementar um sistema de reconhecimento de voz que permita aos professores controlar o projetor inteligente utilizando comandos verbais.
3. **Implementar a conexão com Wi-Fi:** Estabelecer a conexão do projetor inteligente com a rede wi-fi do estabelecimento, permitindo o controle por voz e a atualização de software.
4. **Testar funcionalidades de projeção multimídia:** Garantir que o projetor inteligente possa replicar a tela de um computador, exibindo imagens, vídeos e apresentações de forma clara.
5. **Testar e validar o sistema:** Realizar testes para garantir a funcionalidade e eficiência do projetor inteligente, ajustando e aprimorando o sistema conforme necessário.

4. JUSTIFICATIVA

A escolha do tema para o desenvolvimento de um projetor inteligente foi motivada por um problema frequente enfrentado por muitos professores em sala de aula. Professores de idade avançada ou de menor altura frequentemente têm dificuldade para ligar o projetor, que geralmente encontra-se instalado nos tetos da sala de aula, dificultando o início das aulas e consumindo tempo precioso.

A implementação de um projetor por comandos de voz oferece uma solução prática para esse problema, permitindo que os educadores iniciem suas aulas de maneira eficiente, rápida, e sem esforço físico. Além de resolver essa questão, este projeto também contribui para um ambiente de aprendizagem mais dinâmico e inclusivo, facilitando a integração de IoT na educação.

A escolha da plataforma Arduino para o desenvolvimento do hardware garante a compatibilidade com os componentes de IoT e a flexibilidade necessária para futuras expansões e melhorias. Com isso, espera-se que o projetor inteligente melhore significativamente a experiência de ensino, proporcionando uma ferramenta que apoie os professores e beneficie os alunos.

5. DOCUMENTAÇÃO DE REQUISITOS

Tipo/Código	Requisito	Descrição
RFH01	Reconhecimento de Voz	O sistema deve ser capaz de interpretar comandos verbais
RFH02	Conexão com Wi-Fi	O sistema deve se conectar à rede Wi-Fi da instituição para permitir o envio de comandos
RFH03	Compatibilidade com Arduino	O software deve ser compatível com a plataforma Arduino, garantindo a integração com os componentes físicos do sistema.
RFS01	Projeção Multimídia	O sistema deve permitir iniciar e controlar o projetor existente por comandos de voz, como o zoom e iluminação
RFS02	Facilidade de Instalação	O sistema deve ser simples de instalar e configurar
RNF01	Desempenho	O sistema deve responder a comandos em tempo real
RNF02	Confiabilidade	O sistema deve operar de forma estável, mesmo em ambientes com ruído ou comandos ambíguos, evitando falhas ou acionamentos acidentais
RNF03	Segurança	Apenas usuários autenticados devem poder enviar comandos que alterem o estado do sistema
RNF04	Escalabilidade	A arquitetura deve permitir a adição de novos projetores inteligentes sem grandes alterações no código ou na rede
RNF05	Manutenibilidade	O sistema deve ser bem documentado e seguir boas práticas de versionamento para facilitar manutenção
RNF06	Usabilidade	O sistema deve fornecer feedback de status ao usuário, como visuais (luz LED) indicando o estado atual do projetor.
RNF07	Tolerância a Falhas	Em caso de perda de conexão, o Arduino deve manter o último estado conhecido do projetor e tentar reconectar automaticamente.

6. REQUISITOS FUNCIONAIS

Requisitos Hard

1. **Reconhecimento de Voz:** Implementação de um sistema robusto de reconhecimento de voz para interpretar comandos verbais.
2. **Conexão com Wi-Fi:** Implementação da capacidade de conexão com a rede Wi-Fi da faculdade para permitir comandos de voz e atualizações de software.
3. **Compatibilidade com Arduino:** Garantia de que o software funcione corretamente com a plataforma Arduino e seus componentes.

Requisitos Soft

1. **Projeção Multimídia:** Implementação de funcionalidades para iniciar e controlar o projetor existente por voz e permitir a projeção de conteúdos multimídia, como imagens, vídeos e apresentações.
2. **Facilidade de Instalação:** O software deve ser fácil de instalar e configurar, permitindo que os professores e técnicos de TI configurem o projetor inteligente sem dificuldades.

7. REQUISITOS NÃO FUNCIONAIS

- **Desempenho:** O sistema deve responder a comandos de voz ou entrada remota em tempo real, com tempo de resposta inferior a 1 segundo em 95% das interações. Operações críticas, como ligar/desligar o projetor e ajuste de brilho, devem ser processadas com prioridade.
- **Confiabilidade:** O projetor deve operar de forma previsível e estável, evitando falhas ou acionamentos acidentais, mesmo em ambientes com ruído ou comandos ambíguos.
- **Segurança:** Somente usuários autenticados devem poder enviar comandos que alterem o estado do sistema.
- **Escalabilidade:** A arquitetura deve permitir a adição de novos dispositivos (projetores inteligentes) sem necessidade de grandes alterações na base de código ou na estrutura de rede. O backend deve suportar múltiplas conexões simultâneas e gerenciar diferentes salas/ambientes.
- **Manutenibilidade:** O sistema deve ser modular e documentado, facilitando a manutenção, correção de erros e inclusão de novas funcionalidades. A estrutura do código deve seguir boas práticas de versionamento.
- **Usabilidade:** O projetor deve oferecer feedback sobre seu status (ex: "ligado", "brilho em 70%")
- **Tolerância a falhas:** Em caso de falha na conexão com o servidor, o Arduino deve manter o último estado conhecido do projetor e tentar reconectar periodicamente.

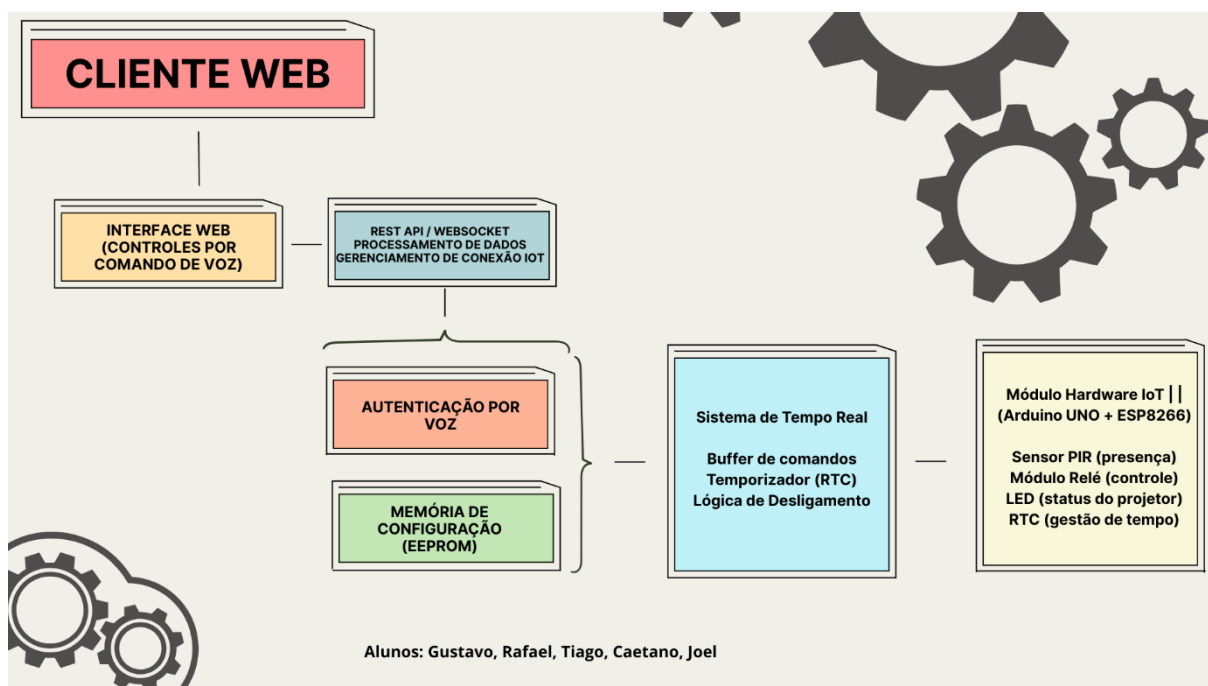
8. ARQUITETURA DO SOFTWARE

- **Modelo Arquitetural:** Cliente-Servidor com IoT
- **Cliente Web (Interface de Controle):** Serve como interface para controle e monitoramento remoto do projetor inteligente.
- **Banco de Dados:** SQLite, utilizado para armazenar dados históricos, configurações personalizadas, estatísticas de uso e logs do sistema na nuvem.
- **Servidor Backend:** Responsável por gerenciar a API REST, autenticação dos usuários e registro de logs de funcionamento.
- **Sistema de Tempo Real:** EEPROM do Arduino: Armazena configurações locais (ex.: tempo de desligamento automático). RTC (Relógio): Gerencia temporizações críticas para desligamento. Memória do ESP8266/ESP32: Cache de comandos recentes para recuperação rápida.
- **Módulo de Hardware (IoT):** composto por Arduino UNO com sensor de presença (PIR), módulo RTC (relógio em tempo real), relé e LED. O Arduino realiza a comunicação com o backend por meio de um módulo Wi-Fi (ESP8266 ou ESP32), permitindo integração com a infraestrutura web.

Componentes utilizados no **Tinkercad**:

- Arduino UNO
- Sensor PIR (detecção de presença)
- Módulo Relé (simula o controle do projetor)
- LED (representa o projetor ligado)
- Botão (acionamento manual)
- RTC (simulado logicamente)
- Resistores e jumpers

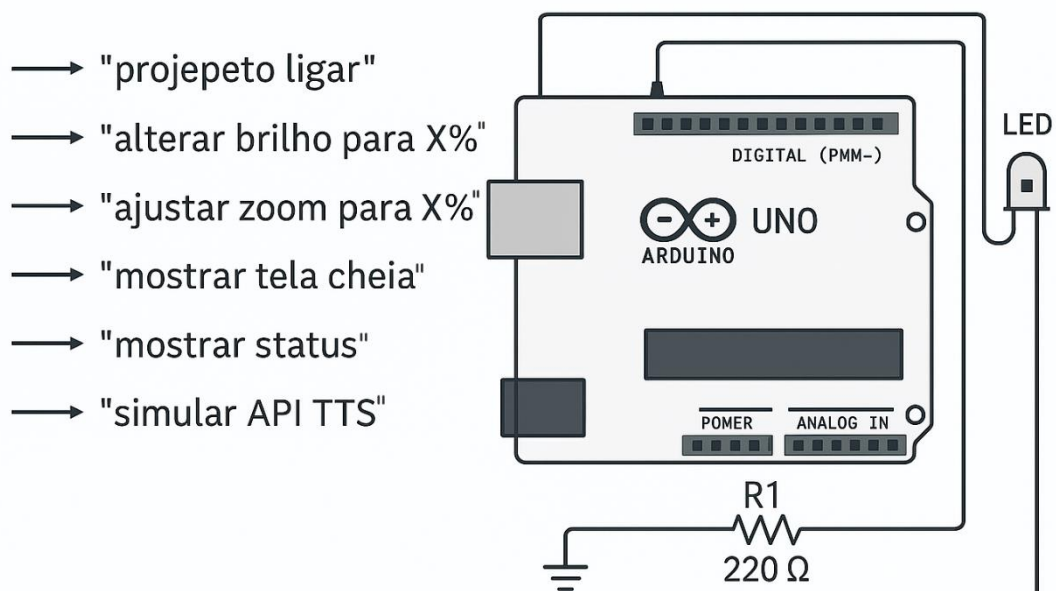
Imagem 1 – Arquitetura de Software do Projetor Inteligente



Fonte: Autoria Própria.

9. DIAGRAMA ELETRÔNICO

Imagem 2 – Diagrama Eletrônico do Projetor Inteligente, com a placa Arduino Uno



Fonte: Autoria Própria

10. CÓDIGO DO PROJETO EM PYTHON

Imagem 3 – Código do projetor inteligente em Linguagem Python

```
import time

ligado = False
brilho = 100
zoom = 0

def atualizar_brilho():
    pwm = int((brilho / 100) * 255)
    print(f"[PWM Simulado] LED com intensidade {pwm} (Brilho: {brilho}%)")

def extrair_porcentagem(frase):
    frase = frase.replace("%", "")
    digitos = ""
    for c in frase:
        if c.isdigit():
            digitos += c
        elif digitos:
            break
    return int(digitos) if digitos else -1

def mostrar_status():
    print("Status:")
    print(f"Brilho: {brilho}%")
    if zoom == 0:
        print("Zoom: Tela cheia")
    else:
        print(f"Zoom: {zoom}%")
    print("Projetor está ligado")

def falar(texto):
    print(f"Enviando para API TTS: '{texto}'...")
    time.sleep(1) # simula tempo da API
    print(f"API TTS: áudio gerado para '{texto}'. (Simulação concluída)")

print("Aguardando comando: 'projepeto ligar' ou 'projetor inteligente opet ligar'")

while True:
    comando = input("> ").strip().lower()

    if not ligado:
        if comando in ["projepeto ligar", "projetor inteligente opet ligar"]:
            ligado = True
            atualizar_brilho()
            print("Projetor ligado!")
            print("Use comandos como:")
            print("'projepeto desligar', 'projepeto mostrar status', 'falar <texto>'")
        else:
            print("Sistema inativo. Diga 'projepeto ligar' ou 'projetor inteligente opet ligar'.")
    else:
        if comando.startswith("projepeto"):
            comando = comando.replace("projepeto", "").strip()
        elif comando.startswith("projetor inteligente opet"):
            comando = comando.replace("projetor inteligente opet", "").strip()
        else:
            print("Comando ignorado. Use 'projepeto' ou 'projetor inteligente opet' antes do comando.")
            continue

        if comando == "desligar":
            ligado = False
            print("Projetor desligado.")
            print("Aguardando comando: 'projepeto ligar'")
        elif comando == "mostrar status":
            mostrar_status()
        elif comando.startswith("alterar brilho para"):
            valor = extrair_porcentagem(comando)
            if 1 <= valor <= 100:
                brilho = valor
                atualizar_brilho()
                print(f"Brilho alterado para {brilho}%.")
            else:
                print("Valor inválido. Use valores entre 1 e 100.")
        elif comando.startswith("ajustar zoom para"):
            valor = extrair_porcentagem(comando)
            if 0 <= valor <= 100:
                zoom = valor
                print(f"Zoom ajustado para {zoom}%.")
            else:
                print("Valor inválido. Use valores entre 0 e 100.")
        elif comando == "mostrar tela cheia":
            zoom = 0
            print("Zoom ajustado para 0%. Tela cheia ativada.")
        elif comando.startswith("falar "):
            texto_para_falar = comando[6:].strip()
            falar(texto_para_falar)
        else:
            print("Comando não reconhecido. Use: desligar, alterar brilho para X%, ajustar zoom para X%, mostrar tela cheia, mostrar status, falar <texto>.")
```

Fonte: Autoria Própria

11. PROJETO CONCEITUAL

Referencial Teórico

“A Internet das Coisas pode auxiliar professores e funcionários a reduzir o esforço e o tempo despendidos com atividades repetitivas.” (Redação Lyceum, 2021).

“O reconhecimento de voz permite multitarefa e conforto com as mãos livres.” (Shaip, 2025).

Metodologia

Desenvolvimento do Sistema:

- **Hardware:** Utilização da plataforma Arduino para criar o hardware necessário para o projetor inteligente.
- **Software:** Desenvolvimento do software de reconhecimento de voz e integração com a rede Wi-Fi do estabelecimento.
- **Testes:** Realização de testes rigorosos para garantir a funcionalidade e eficiência do projetor inteligente.

Implementação:

- **Instalação:** Facilitar a instalação e configuração do sistema para uso em salas de aula.
- **Atualizações:** Implementar um sistema de atualizações automáticas para manter o software atualizado.
- **Compatibilidade:** Garantir que o sistema seja compatível com dispositivos móveis para maior flexibilidade de uso.

12. CRONOGRAMA DE EXECUÇÃO

Etapas	Descrição	Duração	Período
1. Planejamento	Definição dos objetivos, requisitos e metodologia do projeto.	2 semanas	01/03/2025 - 15/03/2025
2. Pesquisa e documentação	Revisão de literatura e elaboração da documentação técnica.	3 semanas	16/03/2025 - 12/04/2025
3. Desenvolvimento de Hardware	Criação do hardware utilizando a plataforma Arduino.	4 semanas	14/04/2025 - 05/05/2025
4. Desenvolvimento de Software	Implementação do sistema de reconhecimento de voz e conectividade Wi-Fi.	4 semanas e 4 dias	05/05/2025 – 06/06/2025
5. Integração e Testes	Integração do hardware e software, realização de testes rigorosos.	1 semana	09/06/2025 - 15/06/2025
6. Ajustes e Melhorias	Ajustes e melhorias com base nos resultados dos testes.	1 semana	16/06/2025 – 22/06/2025
7. Apresentação	Preparação e apresentação do projeto para a banca.	1 dia	23/06/2025

13.TECNOLOGIAS UTILIZADAS

Para o desenvolvimento do projetor inteligente baseado foram utilizadas as seguintes tecnologias:

- **Arduíno UNO R3**

O Arduino UNO R3 é uma placa de prototipagem eletrônica de código aberto baseada no microcontrolador ATmega328P. Ele foi utilizado como o núcleo do sistema embarcado, responsável por receber os comandos de voz processados e acionar o projetor. Sua compatibilidade com diversos sensores e módulos o torna ideal para projetos de automação e IoT.

- **C++**

A linguagem de programação C++ foi utilizada para o desenvolvimento do firmware do Arduino. Por ser a linguagem nativa da plataforma, ela permite o controle direto dos pinos e módulos conectados à placa, além de oferecer desempenho e controle sobre os recursos do microcontrolador.

- **Tinkercad**

O Tinkercad é uma plataforma online de simulação de circuitos eletrônicos. Foi utilizado para simular o funcionamento do sistema antes da montagem física. A ferramenta permitiu testar a lógica de controle, conexões e o comportamento dos componentes de forma prática e segura.

14. CONCLUSÃO E HABILIDADES DESENVOLVIDAS

O desenvolvimento do projetor inteligente demonstrou-se uma solução viável para facilitar o cotidiano dos professores em sala de aula, especialmente aqueles que enfrentam dificuldades para operar projetores instalados em locais de difícil acesso. A proposta de transformar um projetor convencional em um dispositivo inteligente, utilizando tecnologias acessíveis como o Arduino UNO R3, mostrou-se eficaz tanto em termos técnicos quanto educacionais.

Ao longo do projeto, foi possível aplicar conhecimentos teóricos e práticos, além de desenvolver uma série de habilidades essenciais para o trabalho em equipe e para a atuação profissional em áreas de tecnologia e inovação. As principais habilidades desenvolvidas foram:

1. Pensamento Crítico

Durante o projeto, foi necessário analisar diferentes abordagens técnicas, avaliar limitações de hardware e software, e tomar decisões baseadas em testes, evidências e viabilidade prática.

2. Habilidade de Comunicação

A comunicação clara e objetiva entre os membros da equipe foi fundamental para o alinhamento de ideias, divisão de tarefas e resolução de dúvidas ao longo do desenvolvimento.

3. Habilidade de Liderança

A liderança foi exercida na organização das etapas do projeto, na motivação da equipe e na tomada de decisões estratégicas para garantir o progresso contínuo e o cumprimento dos prazos.

4. Habilidade de Planejamento

Foi essencial planejar as fases do projeto, organizar o cronograma de execução, distribuir responsabilidades e prever possíveis obstáculos, garantindo uma execução eficiente.

5. Habilidade de Resolução de Problemas

Diversos desafios surgiram durante o desenvolvimento, como limitações técnicas e ajustes de integração entre hardware e software. A equipe foi capaz de identificar e solucionar esses problemas de forma colaborativa.

6. Habilidade Técnica

O projeto exigiu o domínio de ferramentas e tecnologias como Arduino, linguagem C++ e simulações no Tinkercad. A equipe demonstrou capacidade de aprender e aplicar esses conhecimentos para alcançar os objetivos propostos.

15. LINK DO PROJETO NO TINKERCAD

https://www.tinkercad.com/things/h1dnUR2TD33/editel?returnTo=%2Fdashboard%2Fdesigns%2Fcircuits&sharecode=tdlaKcvvBUFvHeqNlvIR_MGA8s5C5iPx9U7Y_APx09s

16. LINK DO PROJETO HOSPEDADO NO GITHUB

<https://github.com/TheRPessoa/Projeto-Inteligente>

17. REFERÊNCIAS

Lyceum, **Saiba como a internet das coisas pode ser aplicada na educação.**

Redação Lyceum. Disponível em: <https://blog.lyceum.com.br/internet-das-coisas-na-educacao/>. Acesso em: 12 abr. 2025.

Shaip, **O que é reconhecimento de voz: por que você precisa dele, casos de uso, exemplos e vantagens.** Disponível em: <https://pt.shaip.com/blog/voice-recognition-overview-and-applications/>.

Acesso em: 12 abr. 2025.