

Relatório Técnico: EcoFloor - Gestão Energética Inteligente

Inteligência Artificial - FIAP

Grupo 44

Link do Vídeo de Demonstração:

<https://www.youtube.com/watch?v=AQi7sNqF8iw>

Link do Repositório do GitHub:

https://github.com/VictorAraujopy/EcoFloor_GlobalSolution

Solução IoT para Eficiência Energética e Sustentabilidade

Função no Projeto	Nome Completo	RM
Backend & Integração	Victor Araujo Ferreira da Silva	RM567619
Banco de Dados	Jonathan Gomes Ribeiro Franco	RM567109
Hardware & Sensores	Pedro Zanon Castro Santana	RM567350
Data Analytics	Filipe Marques Previato	RM567720
Documentação	Jacqueline Nanami Matushima	RM568498

1. Introdução

O desperdício de energia em ambientes corporativos e educacionais representa um problema significativo, impactando tanto a sustentabilidade ambiental quanto os custos operacionais. O projeto **EcoFloor** surge como uma solução IoT (Internet das Coisas) inteligente, desenvolvida para atuar como um orquestrador que não apenas monitora, mas toma decisões autônomas para otimizar o consumo de energia em salas de aula.

O objetivo principal é demonstrar um sistema robusto, capaz de cruzar dados de **presença** e **temperatura** em tempo real para controlar atuadores (iluminação e ar condicionado), garantindo o conforto do usuário com máxima **eficiência energética** e gerando relatórios auditáveis.

2. Desenvolvimento

O projeto EcoFloor emprega uma arquitetura modular baseada no modelo Cliente-Servidor, integrando hardware simulado, processamento em nuvem e análise de dados. O sistema foi desenhado para garantir a centralização da lógica de negócios e a escalabilidade das conexões.

A. Módulos e Tecnologias Chave:

- **Backend (Servidor API):** Desenvolvido em Python utilizando o framework Flask, atua como o núcleo de processamento do sistema. O arquivo principal, `api.py`, expõe endpoints RESTful para comunicação com os dispositivos IoT, enquanto orquestra a interação entre o banco de dados e o módulo de regras de negócio.
- **Banco de Dados:** Foi utilizada a nuvem Oracle Database para o armazenamento seguro e persistente dos dados. O módulo de conexão, `db_config.py`, implementa um Connection Pool (Piscina de Conexões) para gerenciar múltiplas requisições simultâneas com alta performance. A persistência utiliza instruções SQL otimizadas, como o comando MERGE, assegurando a atualização eficiente do status das salas sem duplicidade de registros.
- **Inteligência de Negócio (Lógica de Decisão):** O módulo `ML.py` contém a lógica de classificação baseada em regras (Rule-Based). Este componente é responsável por identificar padrões de desperdício energético, classificando o estado da sala como "OK", "DESPERDICIO_LUZ" ou "DESPERDICIO_AC" com base nos inputs dos sensores.
- **Hardware (Sensores e Atuadores):** A camada física é simulada através da plataforma Wokwi utilizando a arquitetura do microcontrolador ESP32. O código em C++ coleta dados de presença e temperatura, enviando-os para a API via requisições HTTP POST e consultando comandos de atuação via HTTP GET.
- **Data Analytics e Integração:** Foi desenvolvido o script `gerar_csv.py`, que automatiza a extração do histórico de sensores do banco de dados Oracle, gerando arquivos estruturados para análise estatística e visualização de dados na linguagem R.

B. Fluxo de Decisão (Do Sensor ao Atuador):

O ciclo de automação do EcoFloor segue uma lógica de processamento em tempo real, dividida nas seguintes etapas:

1. **Coleta e Envio:** O dispositivo IoT coleta os dados de presença e temperatura ambiente e os envia para o servidor através da rota `/api/dados_sensor`.
 2. **Processamento e Termostato Inteligente:** Ao receber os dados, a API executa o módulo de classificação e aplica a regra de eficiência energética:
 - Se a temperatura for superior a 23 graus Celsius e houver presença detectada, o sistema define o comando do Ar Condicionado como LIGADO (ON).
 - Se a temperatura for inferior a 23 graus Celsius, o sistema define o comando como DESLIGADO (OFF) para priorizar a economia, independentemente da presença.
 - Se não houver presença detectada, o sistema classifica como desperdício e define todos os comandos (Luz e Ar Condicionado) como DESLIGADO (OFF).
 3. **Persistência:** O resultado do processamento e os dados brutos dos sensores são registrados imediatamente na tabela `logs_sensores` e o estado atualizado da sala é gravado na tabela `status_salas`.
 4. **Atuação:** O dispositivo IoT consulta periodicamente a rota `/api/comandos/sala_id`. A API retorna o estado atualizado (ON ou OFF) calculado no passo anterior, e o hardware executa a ação física correspondente (acionamento de LEDs ou relés).
-

2.2. Análise dos Dados

Lógica de Classificação de Desperdício (ML.py)

O módulo **ML.py** (Inteligência) é a parte mais crítica, pois **SIMULA** o aprendizado de máquina. Ele cruza o estado da sala com a temperatura ambiente para determinar o comando mais eficiente.

Análise e Geração de Relatórios (data/analise/analise_com.R)

O **R** é a ferramenta escolhida para a camada de *Data Analytics*, utilizando os pacotes `dplyr`, `rstudioapi` e `ggplot2` para consumir os dados do servidor e gerar relatórios visuais.

3. Resultados Esperados

O sucesso do projeto é comprovado pela **automação reativa** e pela **comprovação da economia** por meio de dados.

3.1. Demonstração em Vídeo

O vídeo deve demonstrar o ciclo completo do sistema em tempo real:

3.2. Relatório Visual (Gráficos)

Os gráficos gerados pelo R devem claramente evidenciar a diferença entre os cenários. O resultado esperado é um gráfico de linhas onde:

- **Cenário Sem Sistema:** Apresenta uma linha de consumo alta e contínua.
 - **Cenário Com Sistema:** Apresenta picos de consumo seguidos de quedas para zero nos momentos de ausência, provando a eficácia da lógica.
-

4. Conclusões

O projeto EcoFloor demonstrou a viabilidade de uma solução IoT **ponta a ponta** para a gestão energética. A integração bem-sucedida de **Python (Flask)**, **Oracle**, **Wokwi (C++)** e **R (Analytics)** resultou em um sistema funcional que não apenas reage, mas prontamente aplica regras de eficiência.