

## Bibliotecas usadas

A principal funcionalidade deste projeto é a **detecção de plantas** e a identificação do nome de cada uma, com o objetivo de auxiliar no **controle e extermínio de pragas agrícolas**. Para atingir esse objetivo, foi necessário o uso de bibliotecas especializadas que permitem:

1. **Coleta e armazenamento de imagens:** As bibliotecas utilizadas possibilitam a captura de imagens da câmera ESP32-CAM, processamento e envio dessas imagens para análise, utilizando um banco de dados com mais de 10 espécies de plantas.
2. **Execução de inferências com aprendizado de máquina:** A integração com o modelo treinado (usando Edge Impulse) simplifica a classificação das plantas com base nas imagens capturadas, retornando resultados de maneira eficiente.
3. **Código mais enxuto e modular:** O uso dessas bibliotecas reduz a complexidade do código, tornando-o mais legível, reutilizável e fácil de manter. Isso é essencial em projetos de IoT e visão computacional, onde o hardware tem limitações de recursos.

---

## Bibliotecas e suas Aplicações no Contexto do Projeto

- **eloquent\_esp32cam.h e esp\_camera.h:**

Gerenciam a câmera ESP32-CAM para capturar imagens de plantas no campo. Essa funcionalidade é essencial para a criação do banco de dados visual utilizado no projeto.

- **project-1\_inferencing.h:**

Contém o modelo de aprendizado de máquina treinado para identificar espécies de plantas a partir de imagens. Essa biblioteca permite a inferência local, evitando a necessidade de conexões externas durante a execução.

- **edge-impulse-sdk/dsp/image/image.hpp:**

Realiza pré-processamento das imagens, como redimensionamento e normalização, preparando-as para o modelo de inferência. É essencial para lidar com imagens capturadas em condições variáveis no campo.

- **eloquent\_esp32cam/extra/esp32/wifi/sta.h:**

Permite a conexão com redes Wi-Fi, necessária para a comunicação entre o dispositivo

e o banco de dados de imagens, caso seja necessário enviar dados para análise ou atualizar o modelo.

- **WebServer.h:**

Facilita o monitoramento remoto do sistema ao disponibilizar logs do ESP32-CAM via servidor HTTP, acessível pelo navegador. Isso auxilia na depuração e acompanhamento em tempo real.

**A combinação dessas bibliotecas foi essencial para atingir os objetivos do projeto. Elas garantem a captura eficiente de imagens, o processamento necessário e a inferência para identificar plantas, permitindo uma solução prática e confiável para o combate a pragas agrícolas. Além disso, o uso de bibliotecas reduz a necessidade de codificação manual complexa, tornando o código mais enxuto, modular e adaptado às limitações do hardware do ESP32-CAM.**

## **Referências**

Espressif Systems. (2023). **esp\_camera.h Documentation**. Disponível em: <https://github.com/espressif/esp32-camera>.

Edge Impulse. (2023). **Edge Impulse SDK Documentation**. Disponível em: <https://docs.edgeimpulse.com>.

Eloquent Arduino. (2023). **eloquent\_esp32cam.h Documentation**. Disponível em: <https://eloquentarduino.github.io>.

Arduino. (2023). **WebServer.h Documentation**. Disponível em: <https://www.arduino.cc>.