

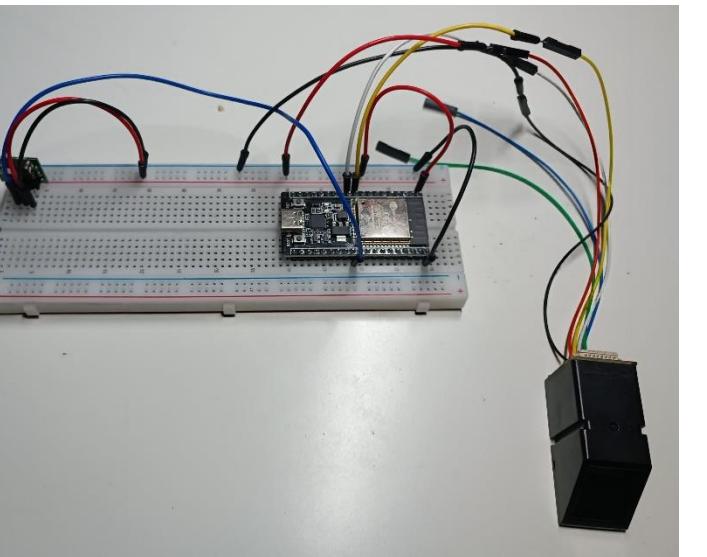
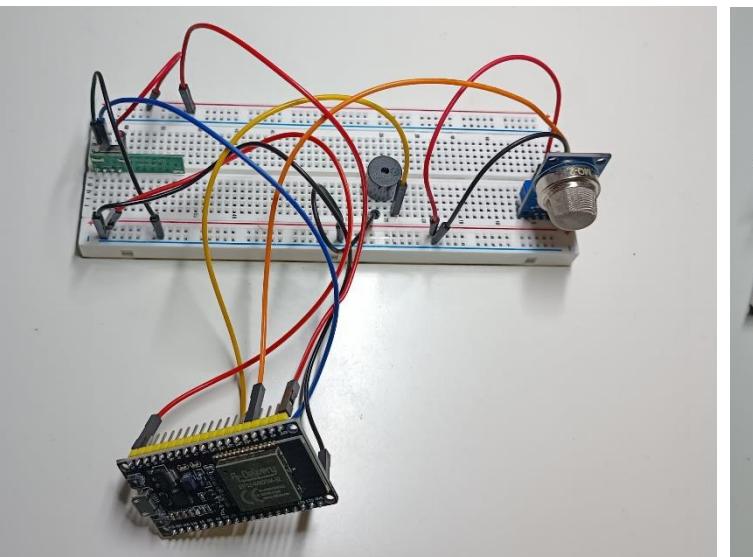


# Trabajo Fin de Máster

*Desarrollo de un sistema de seguridad inteligente integrado con reconocimiento facial, detección de gases y monitoreo de aperturas de puertas y ventanas*

Presentado por **Javier Piay Rodríguez**

Ingeniero en Tecnologías Industriales



# Resumen

Los contenidos principales de esta presentación

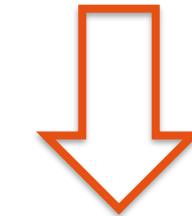
- |           |   |           |                          |
|-----------|---|-----------|--------------------------|
| <b>01</b> | Introducción                            | <b>07</b> | Microcontrolador         |
| <b>02</b> | Objetivo principal                      | <b>08</b> | Arduino Cloud            |
| <b>03</b> | Reconocimiento facial                   | <b>09</b> | Sistema                  |
| <b>04</b> | Alarma de gases                         | <b>10</b> | Resultados               |
| <b>05</b> | Control de apertura de puertas/ventanas | <b>11</b> | Conclusiones             |
| <b>06</b> | Control de acceso dactilar              | <b>12</b> | Cronograma y presupuesto |

# Introducción

1

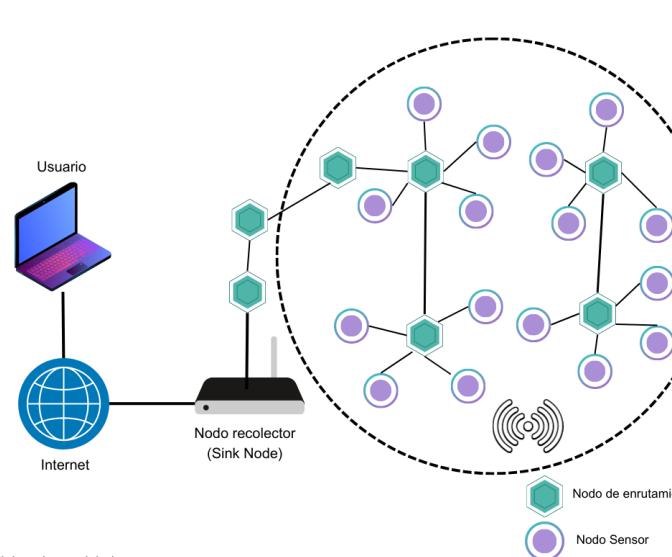
**ORIGEN**

MÁSTER EN INDUSTRIA 4.0		SECURITAS DIRECT		
Internet of Things (IoT)	Inteligencia Artificial (IA)	Preocupación	Robos	Puerta blindada
		93%	71%	+ 50%

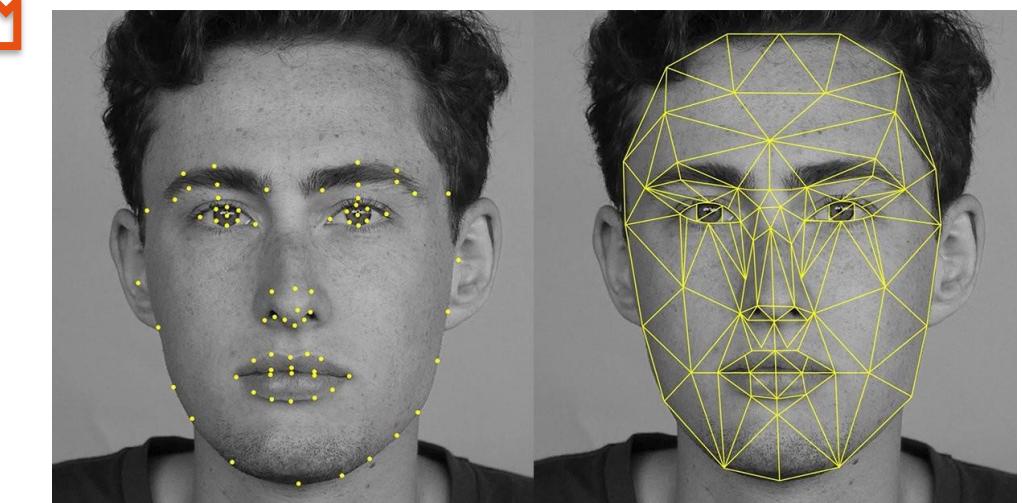


**Desarrollo de un sistema de seguridad**

Red de sensores



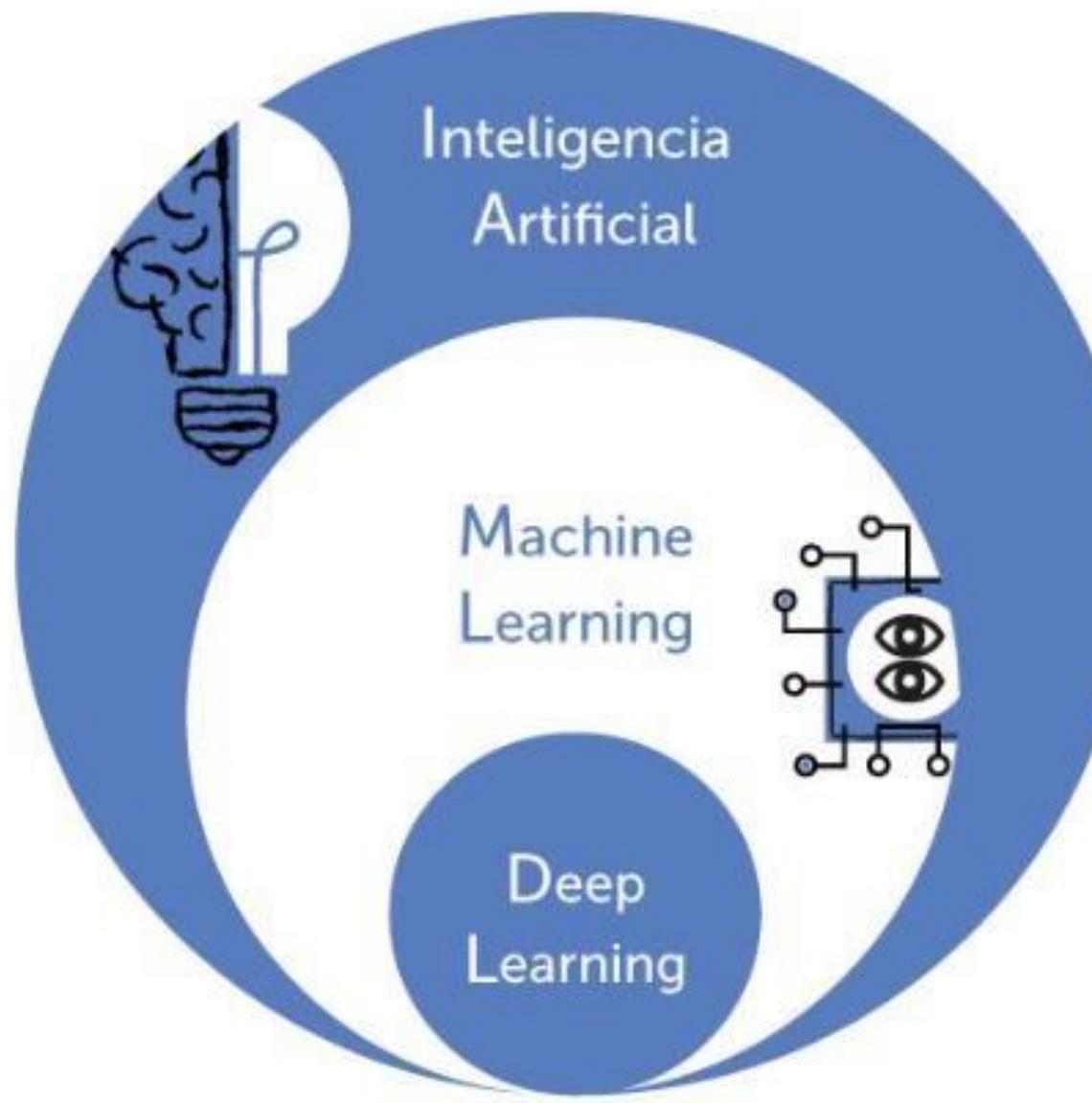
Reconocimiento facial



# Introducción

1

## ASPECTOS CLAVE

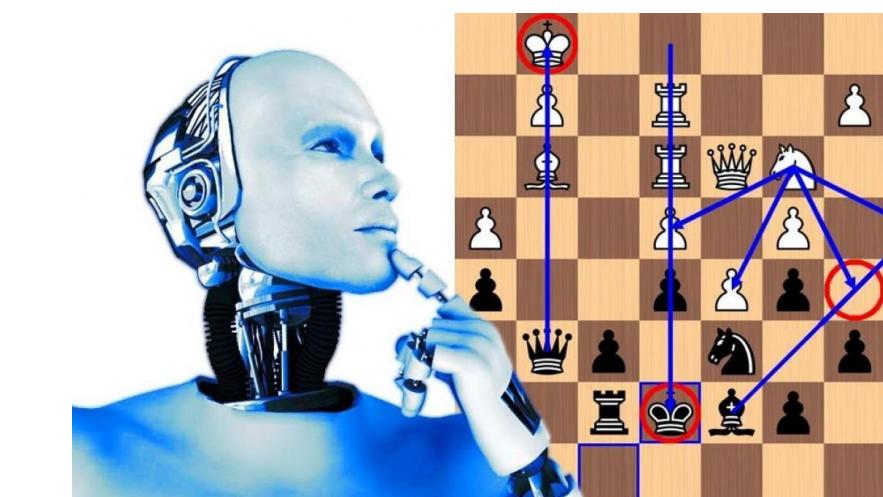


### Inteligencia Artificial

Sistemas y máquinas capaces de realizar tareas que requieren inteligencia humana

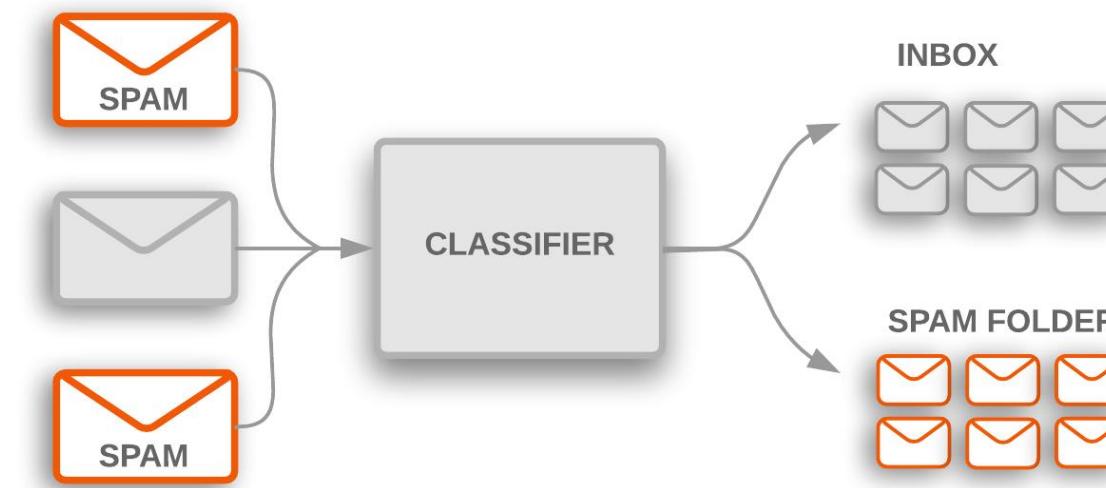
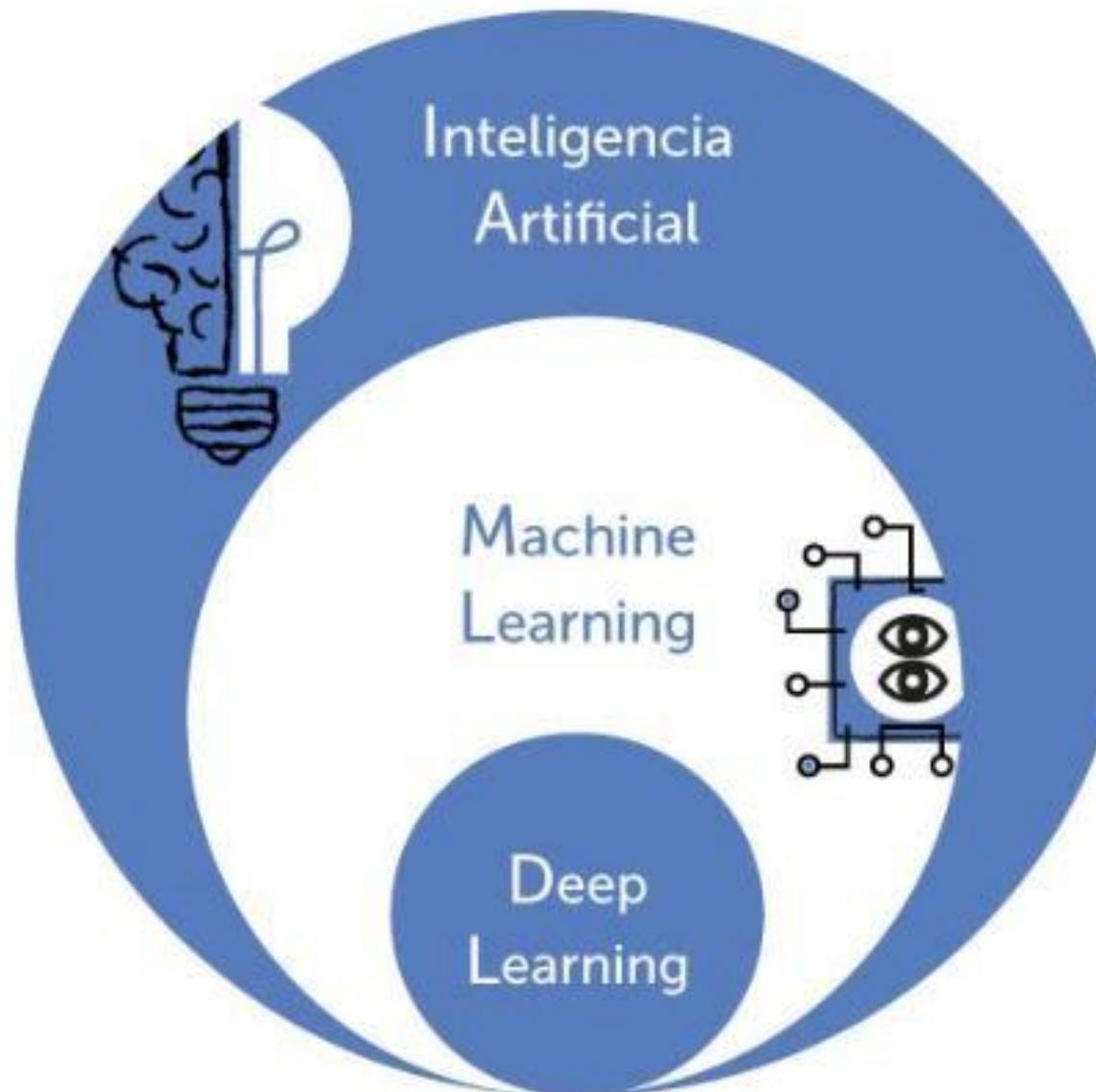


Razonamiento, aprendizaje, percepción o toma de decisiones



# Introducción

## ASPECTOS CLAVE



### Machine Learning

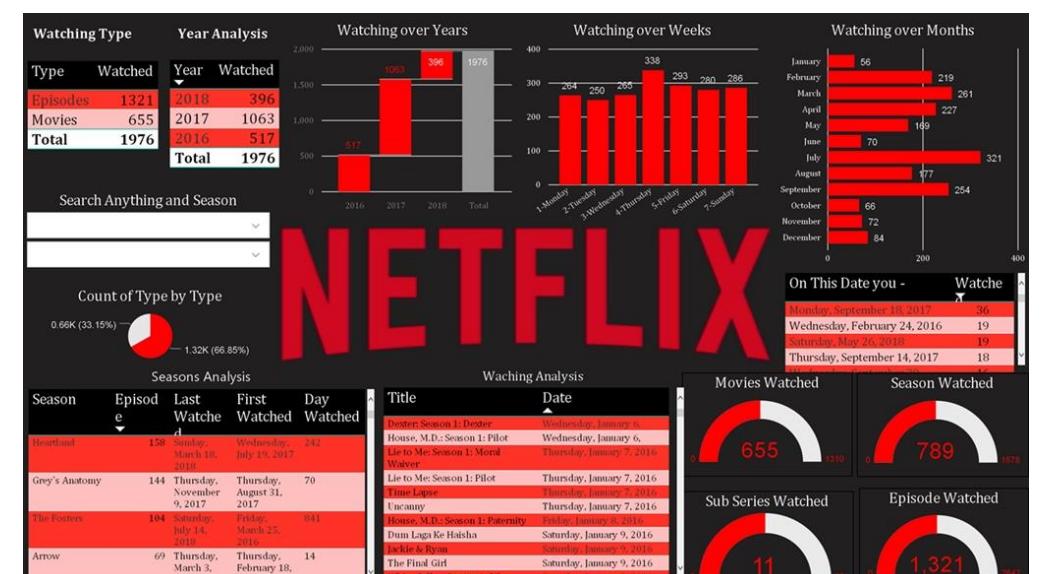
Diseñar algoritmos y modelos



Identificar patrones y tomar decisiones



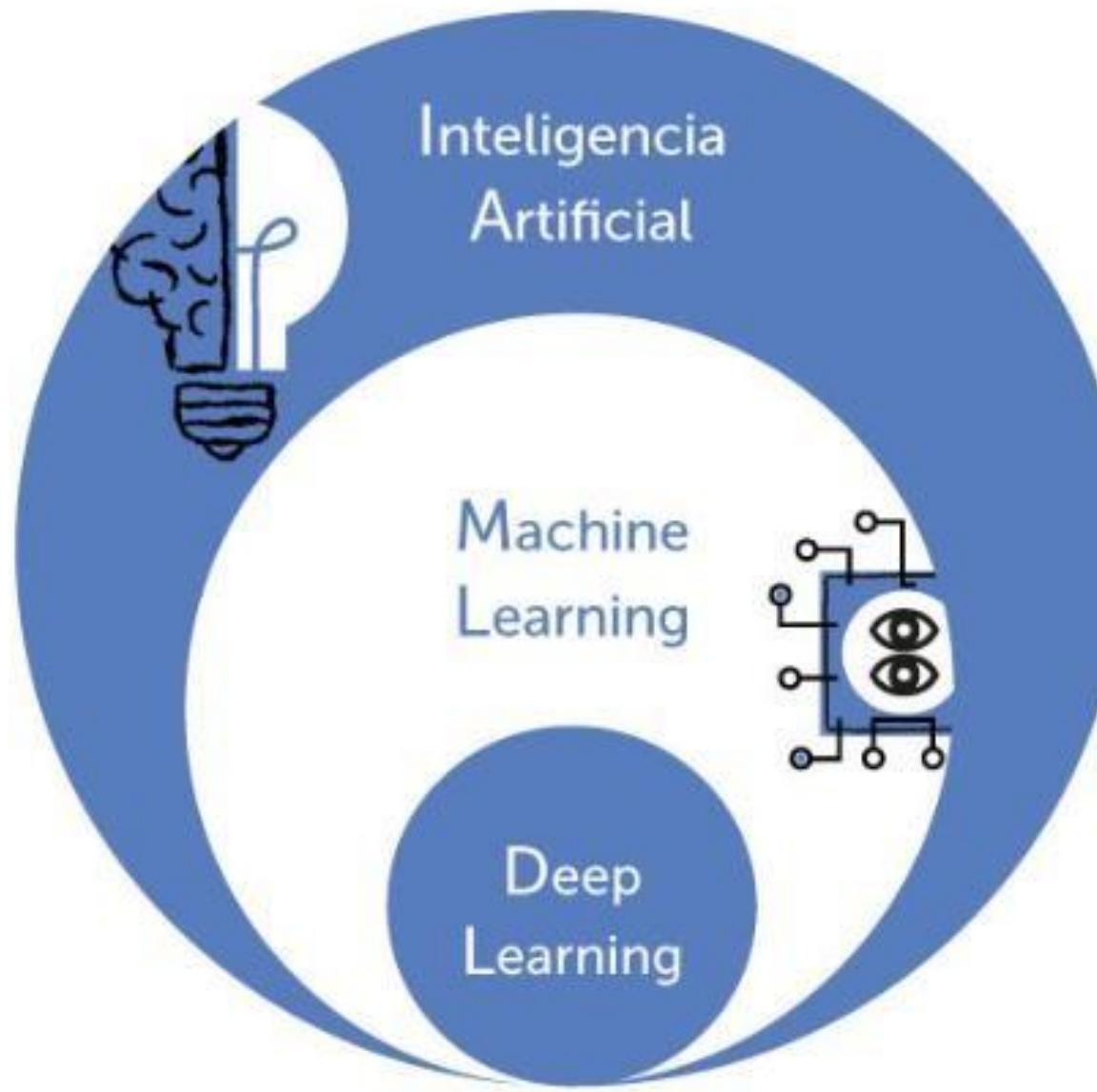
Aprender de datos



# Introducción

1

## ASPECTOS CLAVE

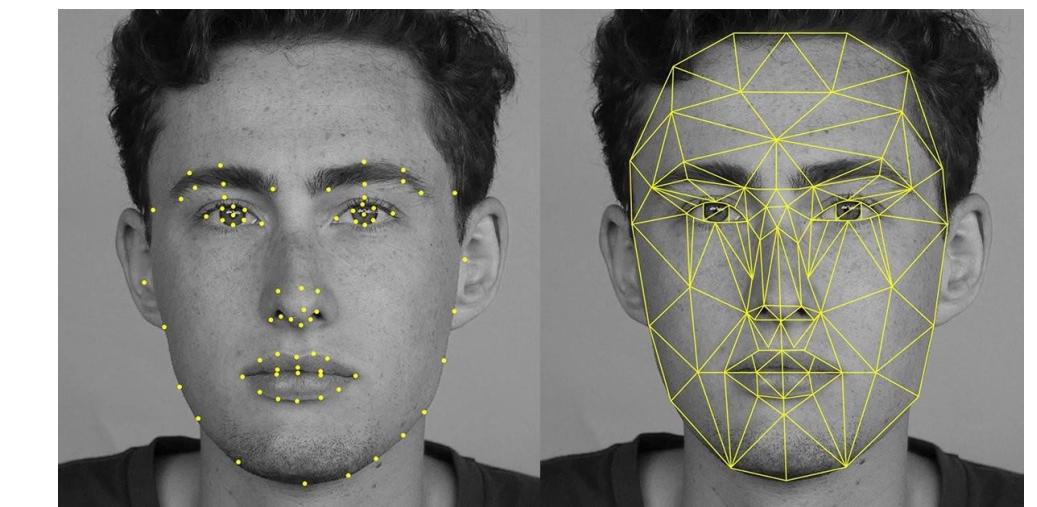
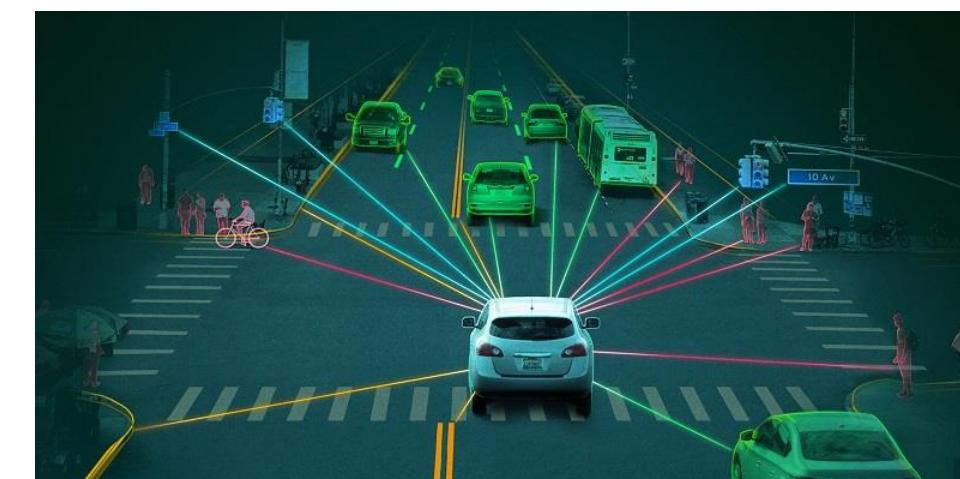


### Deep Learning

**Redes neuronales artificiales (ANNs)** para modelar problemas complejos.



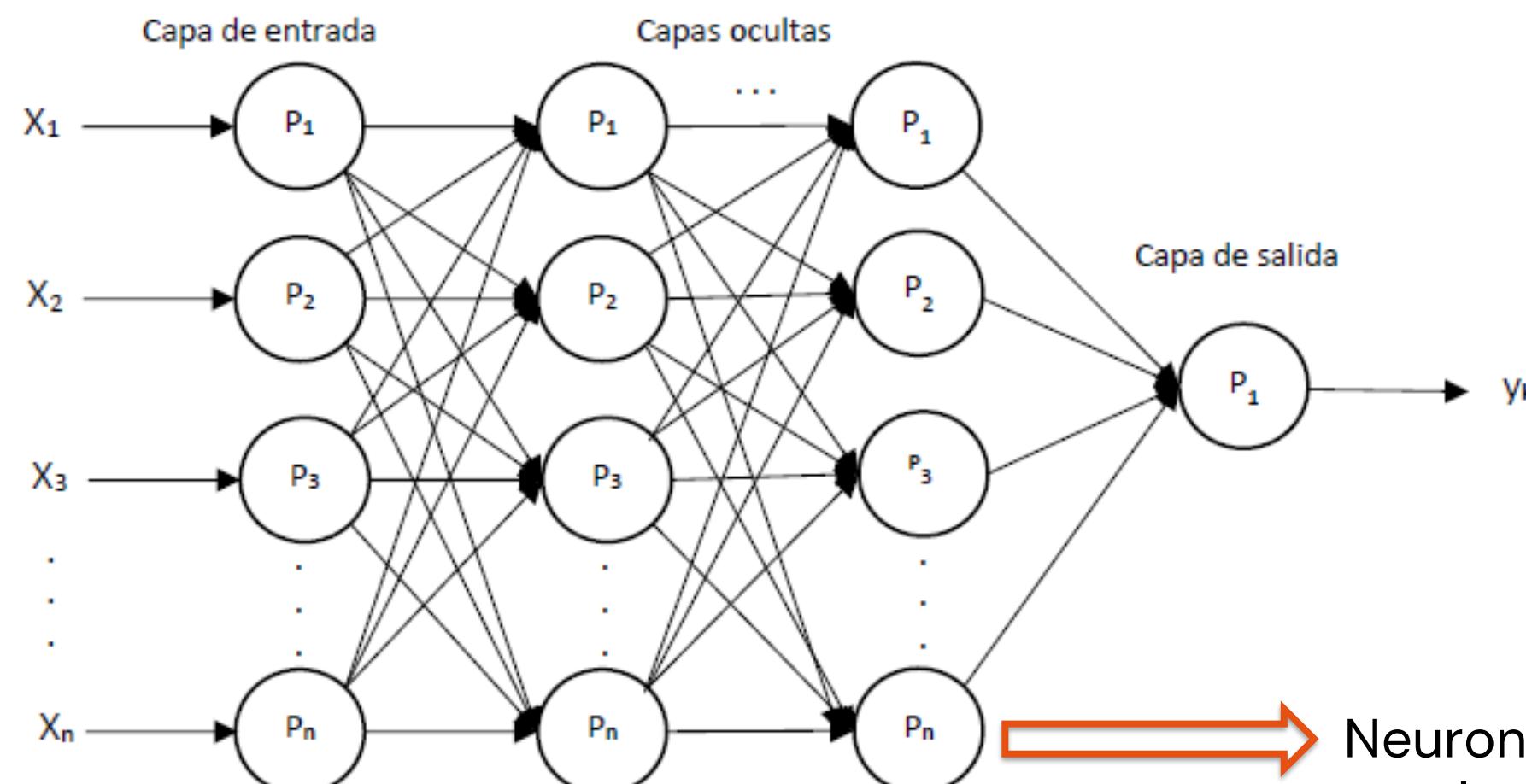
Imitan la estructura del cerebro humano para procesar grandes cantidades de datos y aprender representaciones complejas de forma jerárquica.



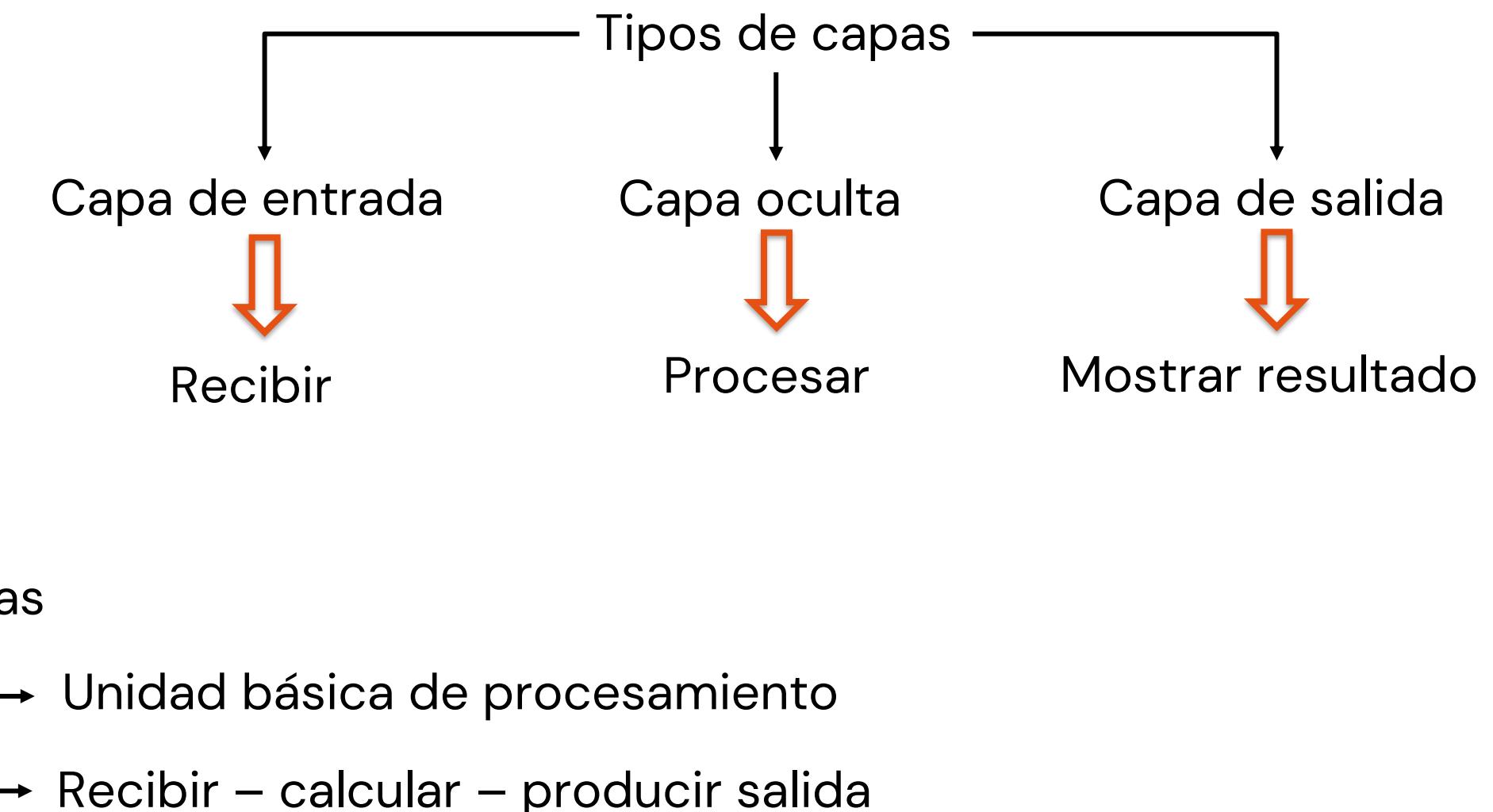
# Introducción

1

## ASPECTOS CLAVE



### Redes neuronales



# Introducción

# ASPECTOS CLAVE



# Internet of Things

# Red de dispositivos físicos conectados a internet

# Recopilar, intercambiar y procesar datos

# Sensores, software y otras tecnologías

# Hogar Inteligente



# Agricultura

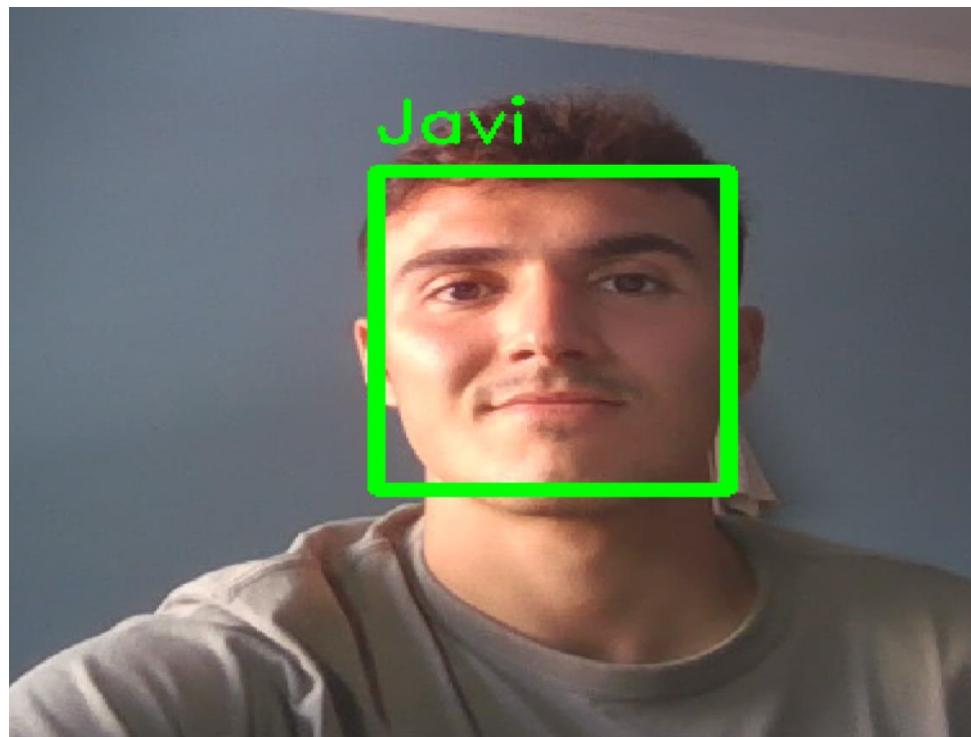


# Objetivo

2

## SISTEMA DE SEGURIDAD INTELIGENTE Y MULTIFUNCIONAL

Reconocimiento facial



Alarma de gases



Control de apertura de puertas y ventanas



Control de acceso dactilar

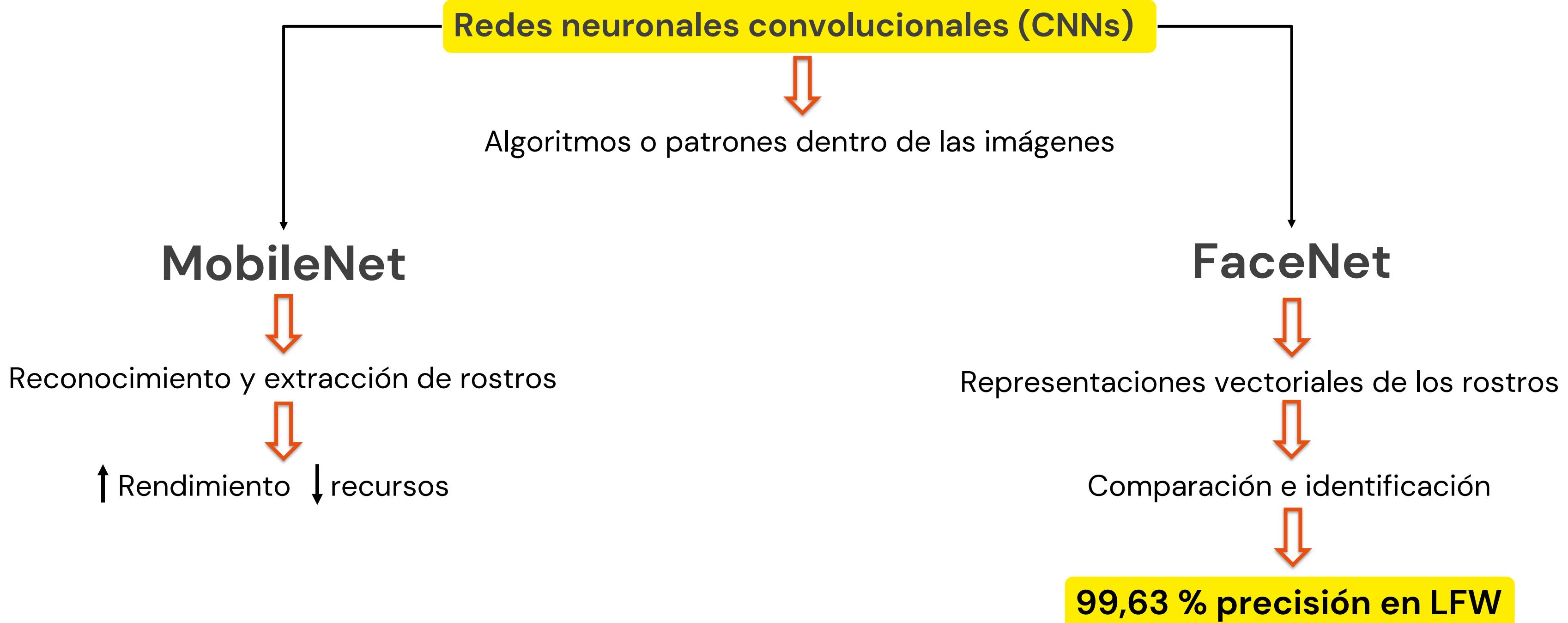


MONITOREO EN TIEMPO REAL

# Reconocimiento facial

3

## REDES NEURONALES



# Reconocimiento facial

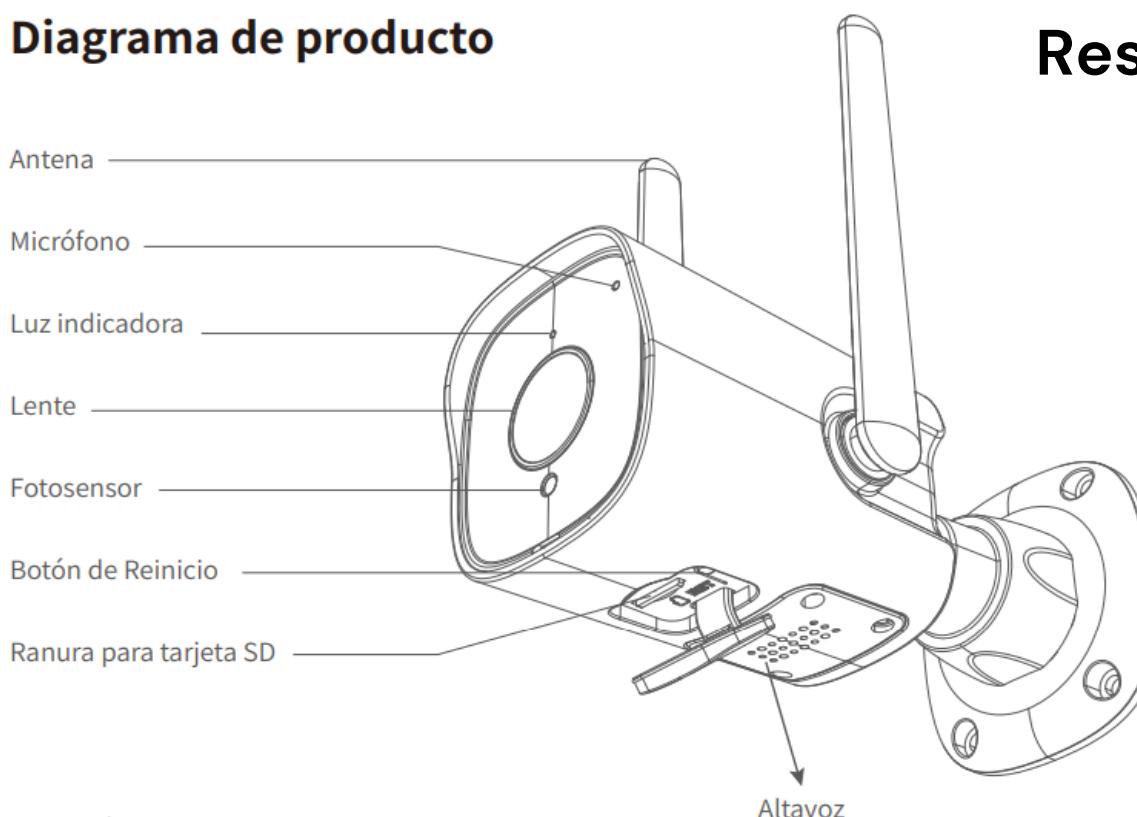
3

## CÁMARA DE VIGILANCIA

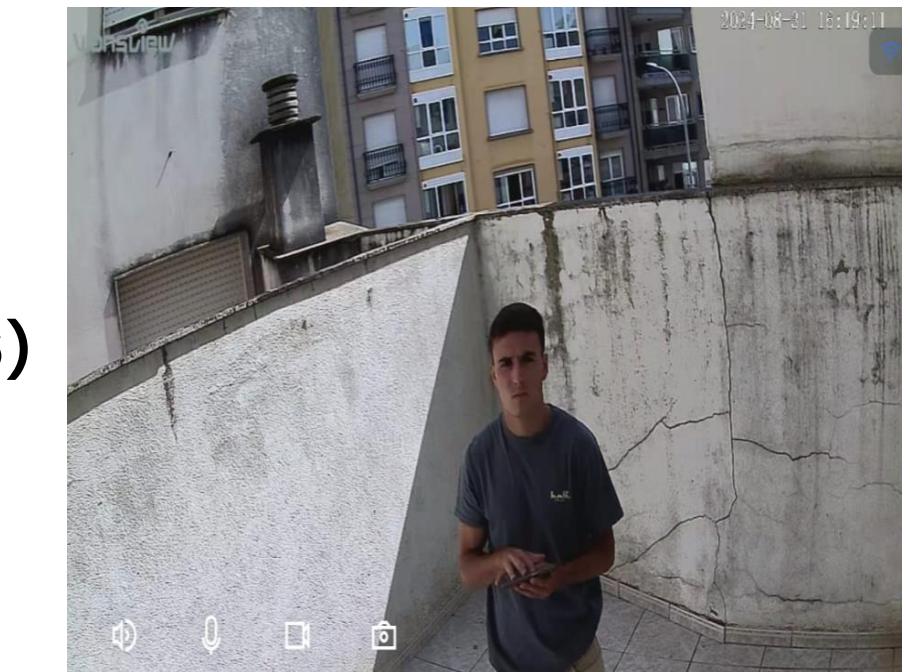
### Wansview Y1



Diagrama de producto



Resolución 2K (2304x1296)



Visión nocturna



Conexión Wi-Fi

RTSP

Wansview Cloud

# Reconocimiento facial

3

**CÓDIGO**

Lenguaje de programación



Entorno de desarrollo

**Bibliotecas**

- Procesamiento de imágenes
- Visión por computadoras



- Aprendizaje automático
- Redes neuronales profundas



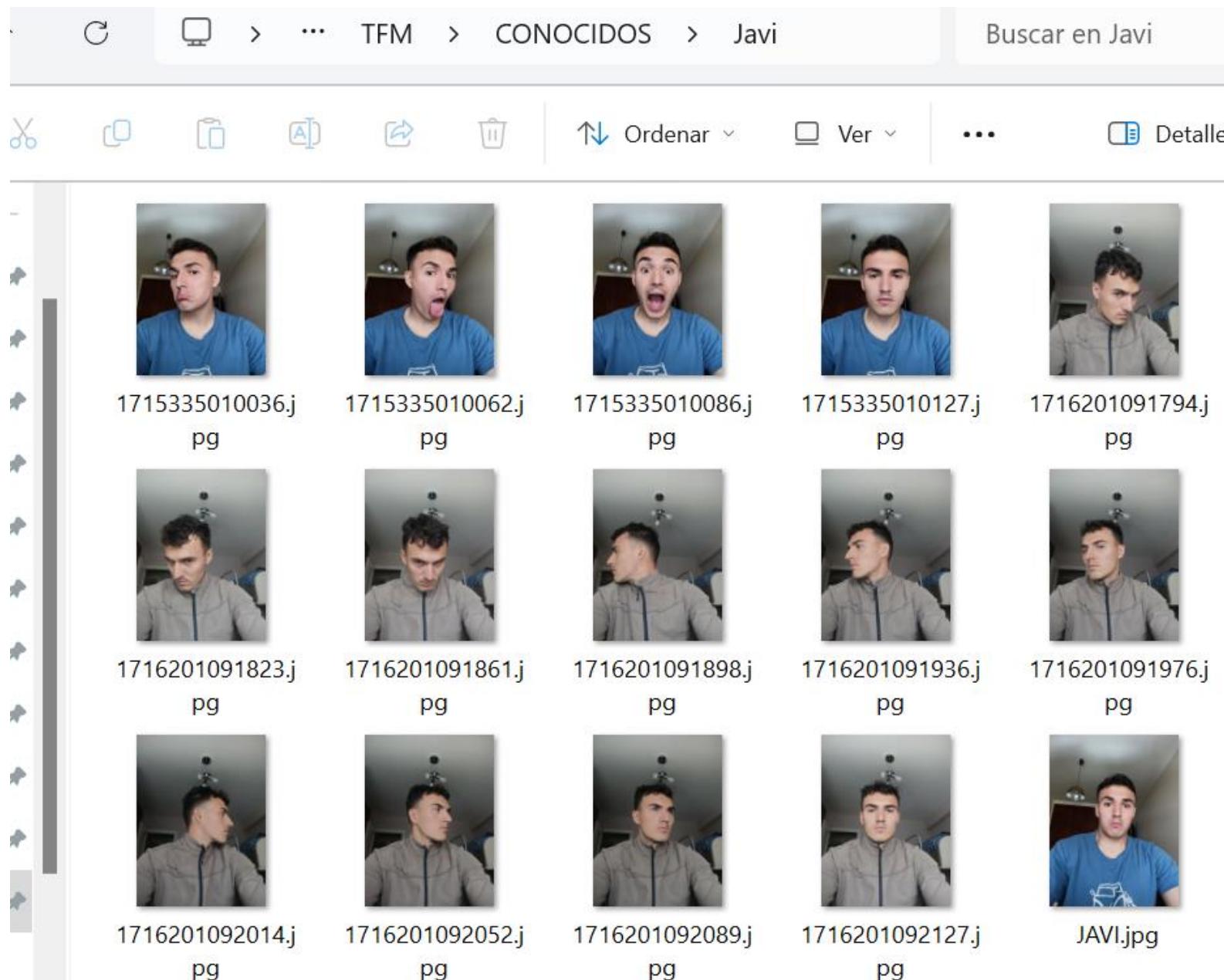
- Computación científica

# Reconocimiento facial

3

# FUNCIONAMIENTO

# **BASE DE DATOS = “CONOCIDOS”**



## **INICIALIZAR MOBILENET**

```
with tf.io.gfile.GFile(r'C:\Users\javiel\pythonOpenCV\TFM\frozen_inference_graph_face.pb','rb') as f:  
    graph_def = tf.compat.v1.GraphDef()  
    graph_def.ParseFromString(f.read())  
  
with tf.Graph().as_default() as mobilenet:  
    tf.import_graph_def(graph_def,name='')  
  
# Inicializar mobilenet  
    sess = tf.compat.v1.Session(graph=mobilenet)  
    image_tensor = mobilenet.get_tensor_by_name('image_tensor:0')  
    boxes = mobilenet.get_tensor_by_name('detection_boxes:0')  
    scores = mobilenet.get_tensor_by_name('detection_scores:0')
```

# Reconocimiento facial

3

**MOBILENET**

Imagen Original



Imagen con Box



Rostro Extraído

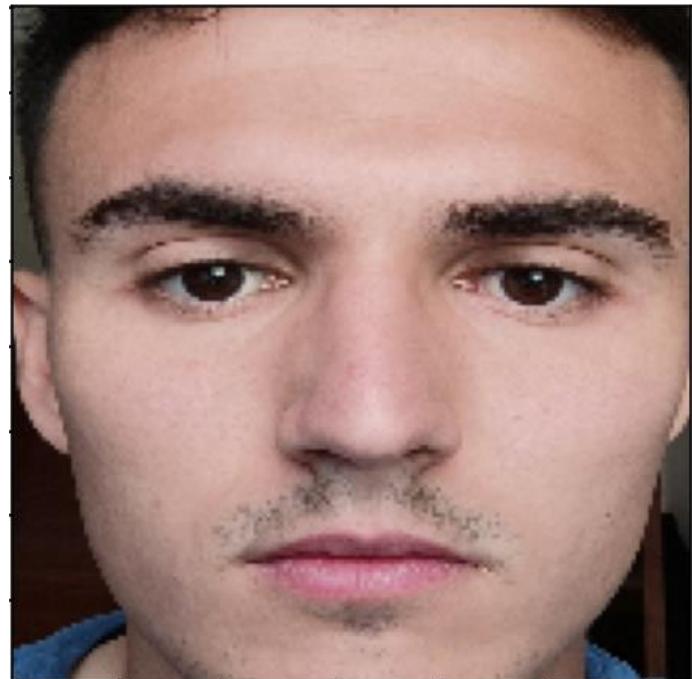


# Reconocimiento facial

3

**FACENET**

Rostro Extraído

**EMBEDDING**

Vector de características

```
[[-0.06862201  0.00066532  0.03701193  0.081644   0.04869827 -0.02080692
 -0.02783912  0.04892395  0.05294482 -0.03024396 -0.05064935 -0.03082394
 0.0041972   0.09105777 -0.04619575 -0.01012124 -0.0251629   0.05916639
 -0.00653461  0.00855683 -0.01221845 -0.02052104 -0.06945418 -0.06682525
 -0.00227261 -0.04024568  0.02510316  0.0424457   0.09479465 -0.08320663
 -0.00571022 -0.00083636  0.06018968 -0.03957112  0.03807572  0.00687763
 -0.0360774   -0.02368525 -0.02481877  0.01299677 -0.02089445  0.02124918
 0.02116491 -0.06147437  0.01130965 -0.04193525  0.03083625  0.04440161
 0.01565098 -0.01459044  0.00077683  0.07255124  0.1096416   0.04185458
 -0.02944057 -0.01076597 -0.00880739  0.07310814 -0.01274593 -0.11524312]
```



Rostro →

MobileNet →

FaceNet →

 $d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$ 

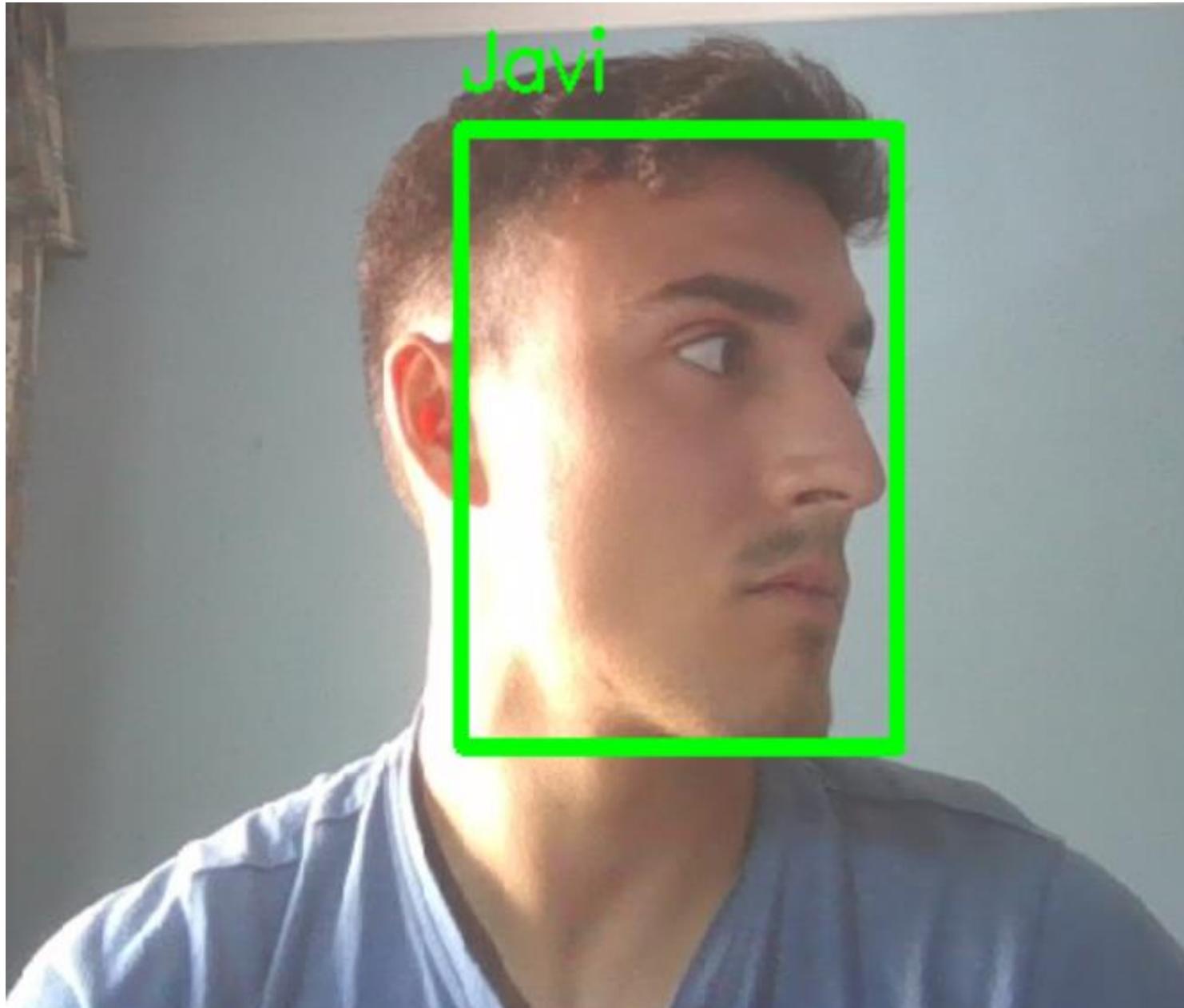
Distancia euclídea

$$\sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

> umbral  
≤ umbral

# Reconocimiento facial

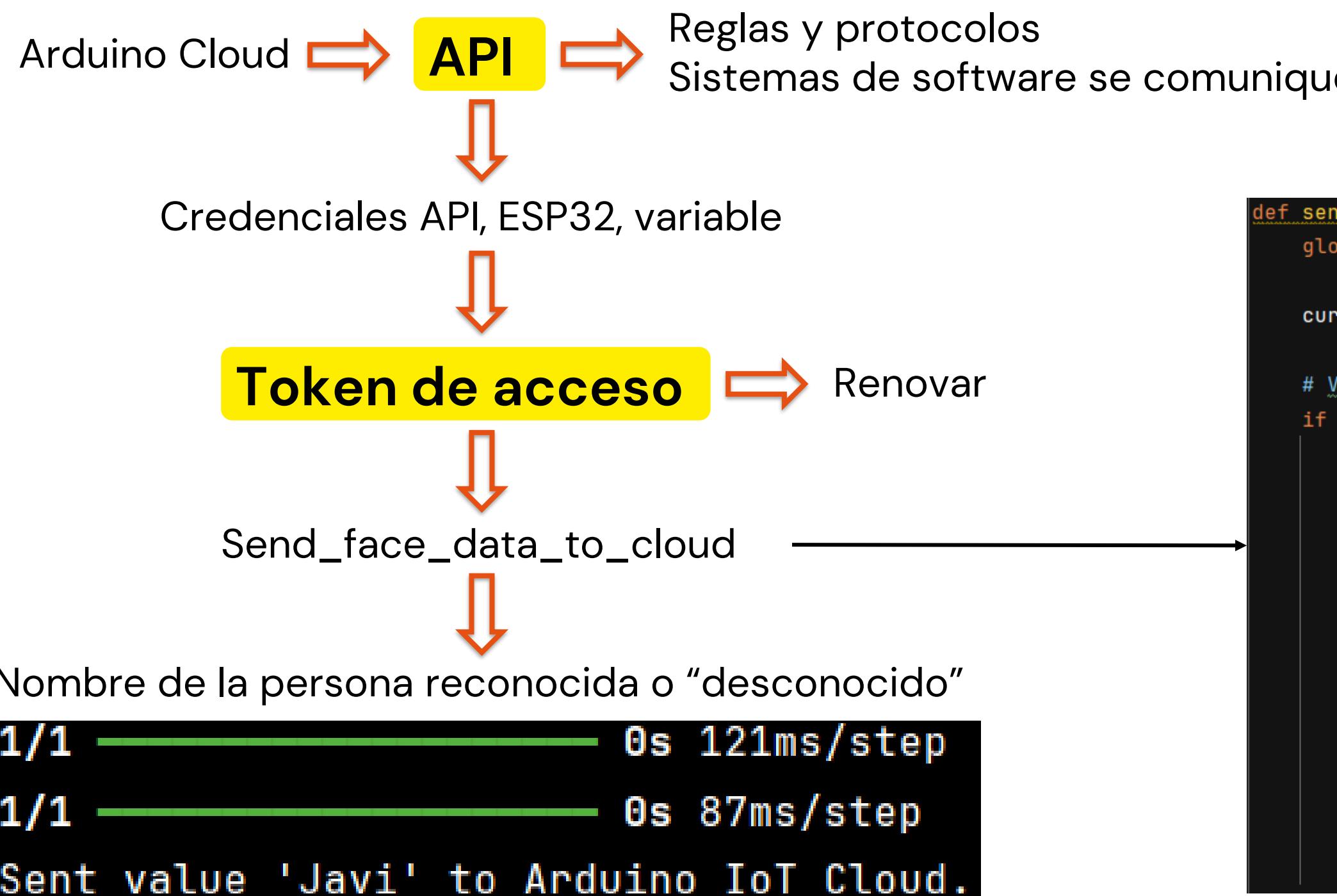
3

**RESULTADO****≤ UMBRAL****> UMBRAL**

# Reconocimiento facial

3

## ENVÍO DE DATOS



```
def send_face_data_to_cloud(value):
    global last_sent_time, MIN_TIME_INTERVAL

    current_time = time.time()

    # Verificar si ha pasado suficiente tiempo desde el último envío
    if current_time - last_sent_time >= MIN_TIME_INTERVAL:
        try:
            properties_api.properties_v2_publish(
                THING_ID,
                PROPERTY_ID,
                property_value: {"value": value}
            )

            print(f"Sent value '{value}' to Arduino IoT Cloud.")

        # Actualizar el tiempo del último envío
        last_sent_time = current_time

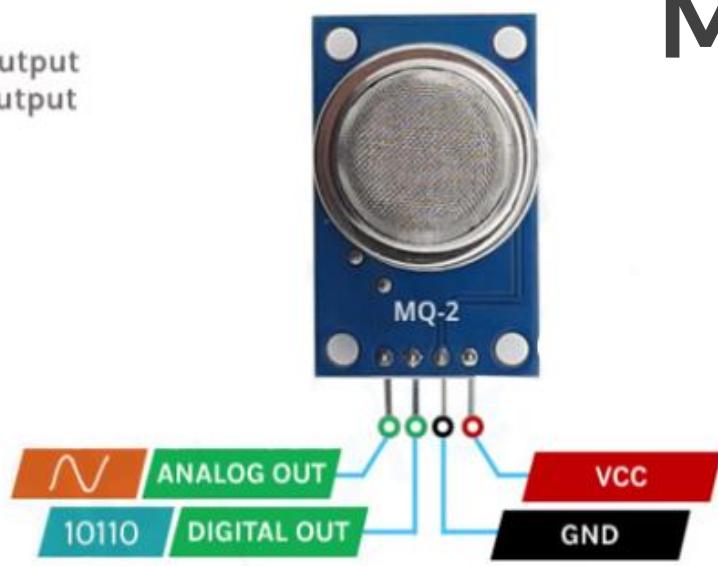
    except ApiException as e:
        print(f"Exception when sending data to Arduino IoT Cloud: {e}")
```

4

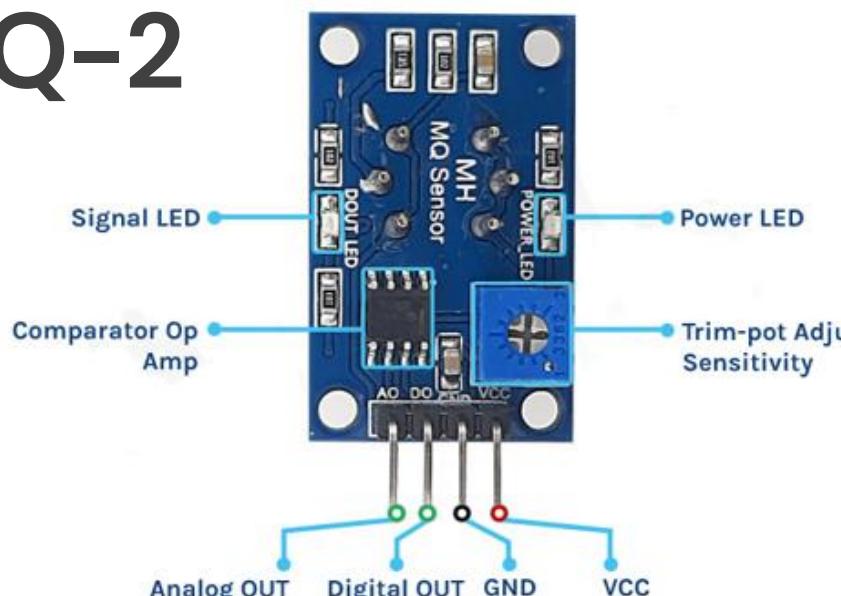
# Alarma de gases

- ❖ Sensor MQ-2
- ❖ Zumbador

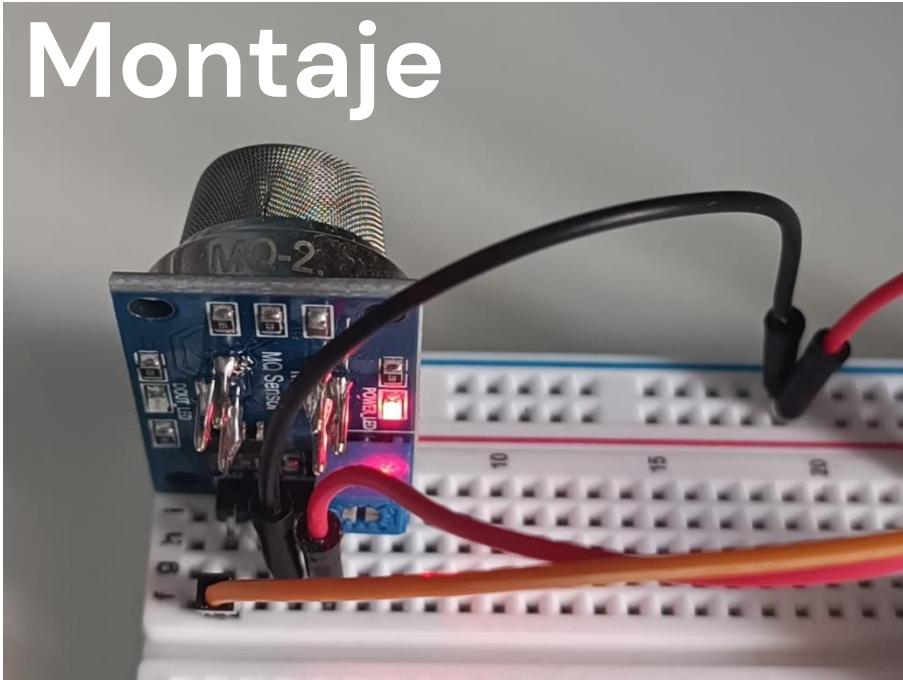
Legend:  
Power  
GND  
Analog Output  
Digital Output



MQ-2



Humo, butano, propano



Montaje

Elemento calefactor



Elemento sensor



Voltaje medible

Zumbador



Montaje

Oscilador interno



Voltaje



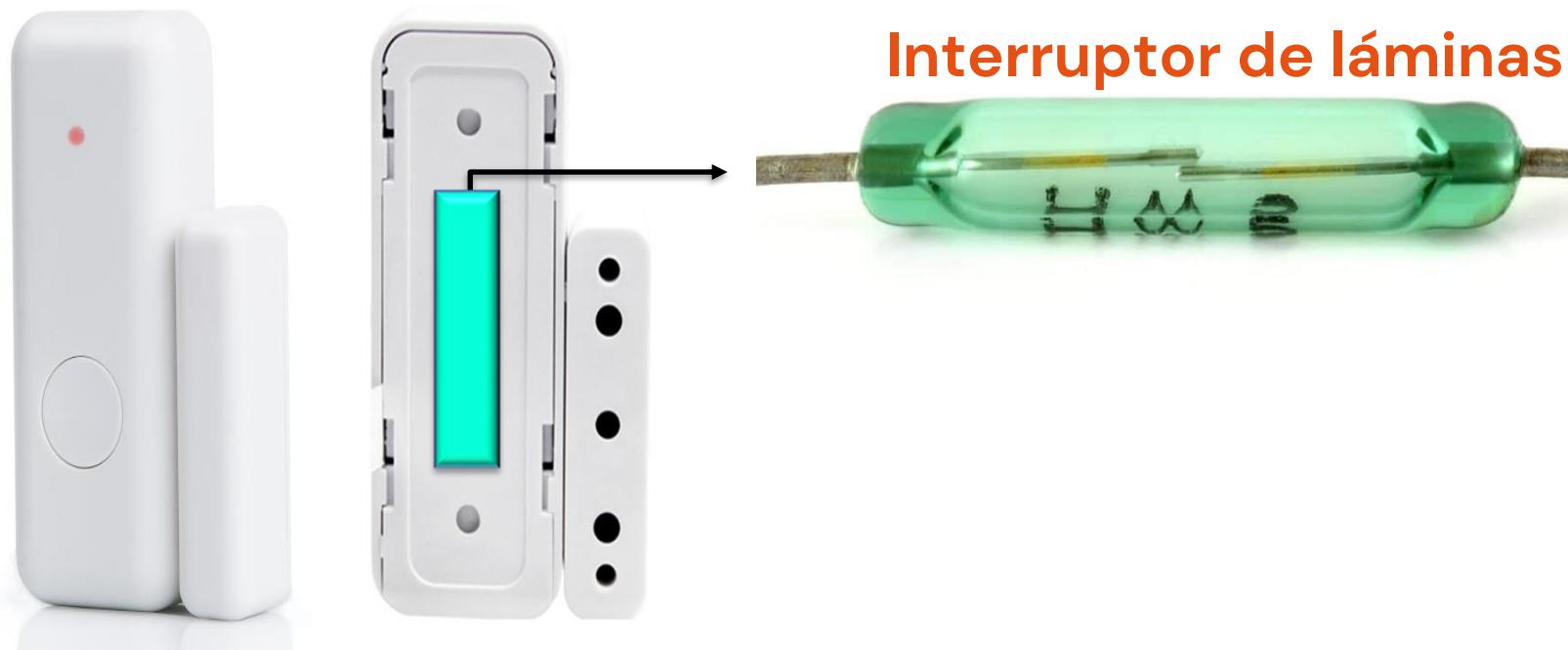
Sonido



# Control de apertura de puertas y ventanas

5

- ❖ Sensor magnético inalámbrico
- ❖ Módulo de radiofrecuencia

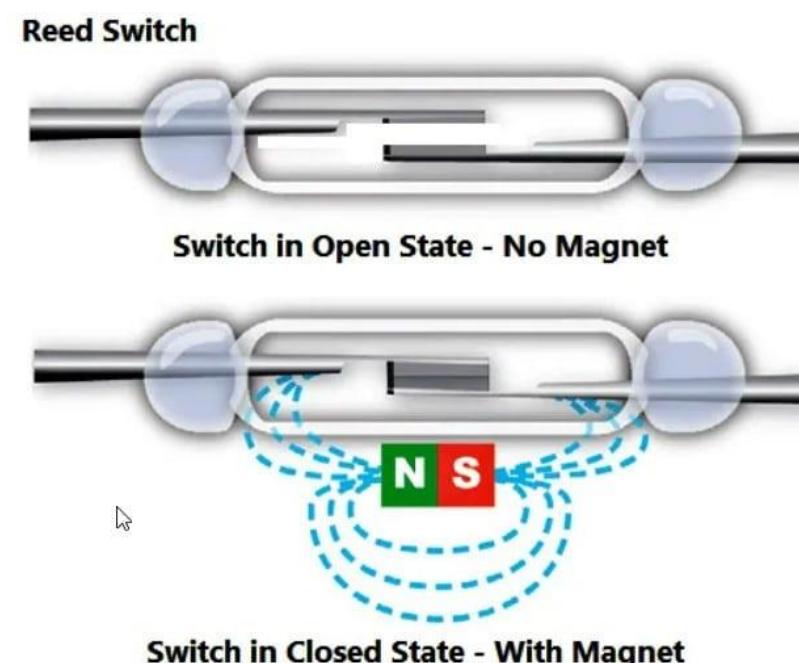


Interruptor de láminas

433 MHz



Imán

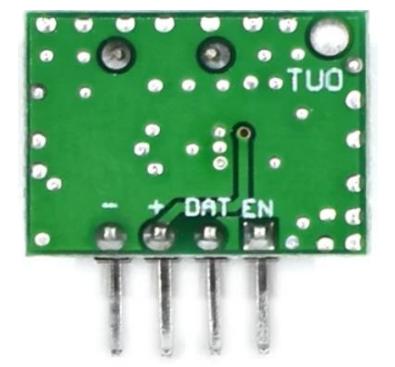
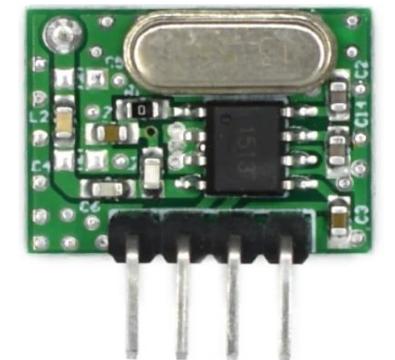
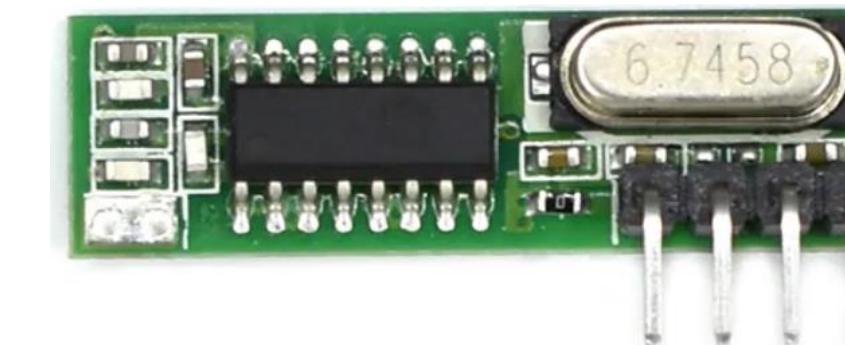


Reed Switch

Switch in Open State - No Magnet

Switch in Closed State - With Magnet

Receptor  
Transmisor



Campo magnético

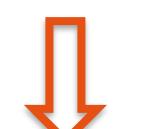


Apertura puerta



Campo magnético

Emite señal de RF



Receptor

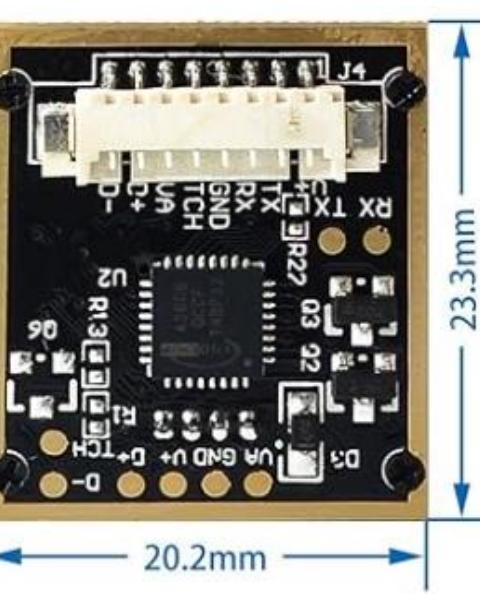
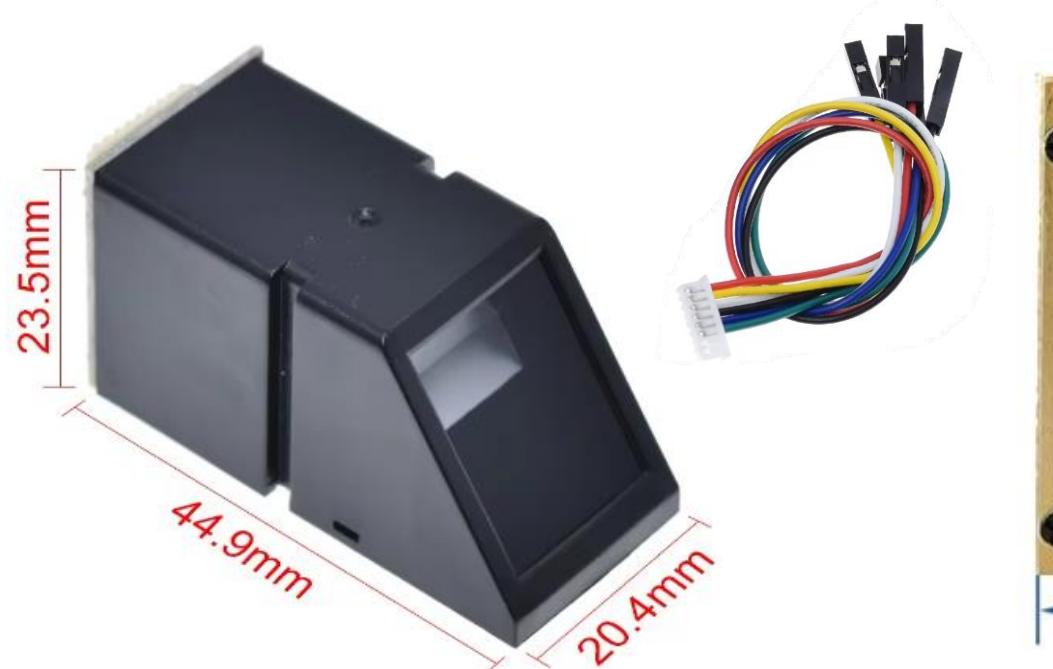


ESP32

# Control de acceso dactilar

6

## Lector de huellas dactilares



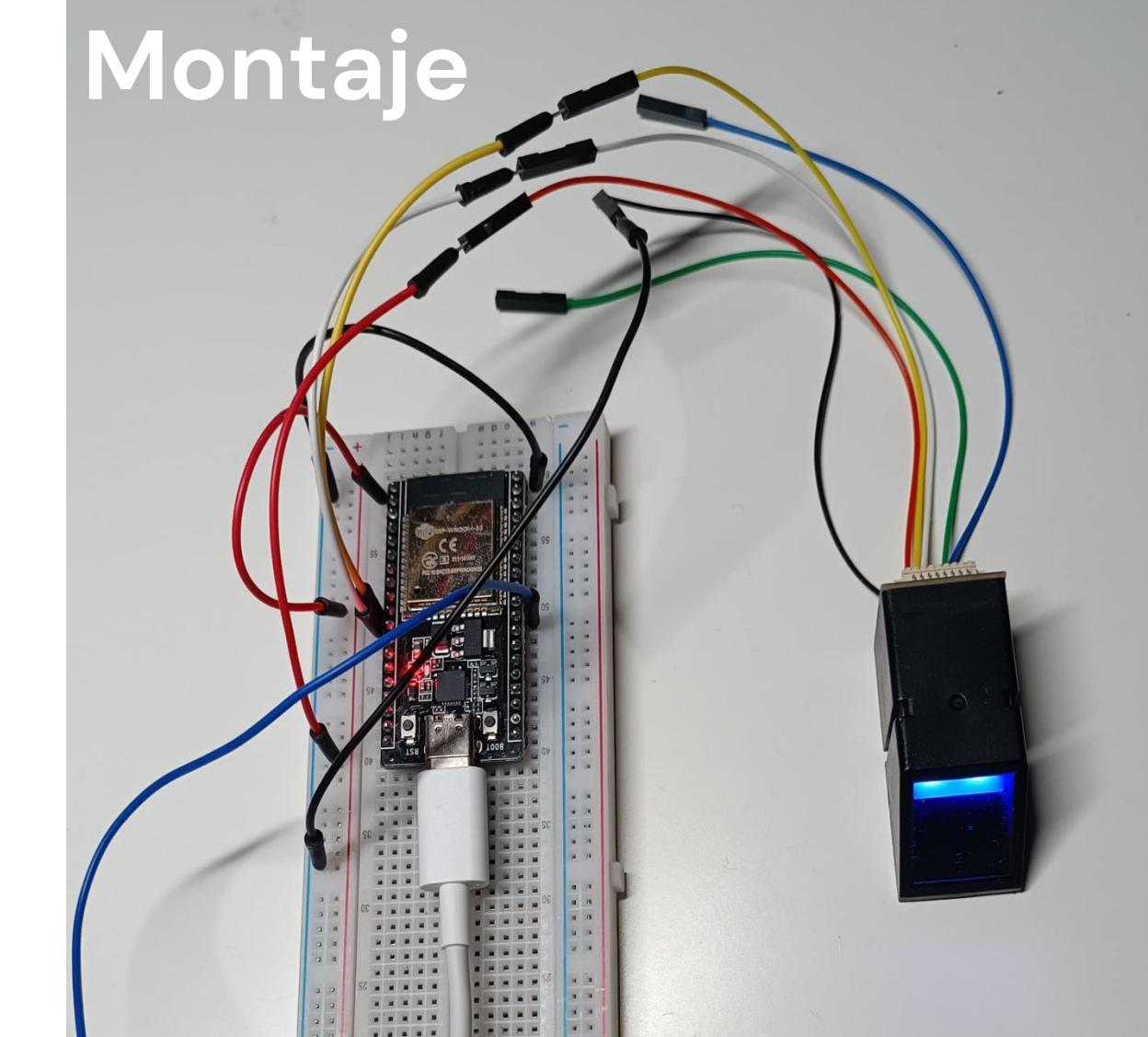
Modelo óptico  
AS608

↓  
Arduino

↓  
Registrar huellas

↓  
Control en  
tiempo real

Salida Monitor Serie x  
Mensaje (Intro para mandar el mensaje de 'NodeMCU-32S' a 'COM7')  
  
Ingrese un ID para la nueva huella dactilar: Registrando ID 4  
Coloque su dedo en el sensor...  
No hay dedo detectado  
Imagen tomada  
Imagen convertida  
Quite su dedo  
Coloque el mismo dedo nuevamente  
Imagen tomada  
Imagen convertida  
Modelo creado  
Huella dactilar registrada exitosamente!



Coloca el dedo → LED ilumina la huella → Patrón de reflexión → Sensor de imagen → Huella dactilar

# Microcontrolador

7

## ESP32 PRINCIPAL

Unidad fundamental de procesamiento

### ESP32 Principal

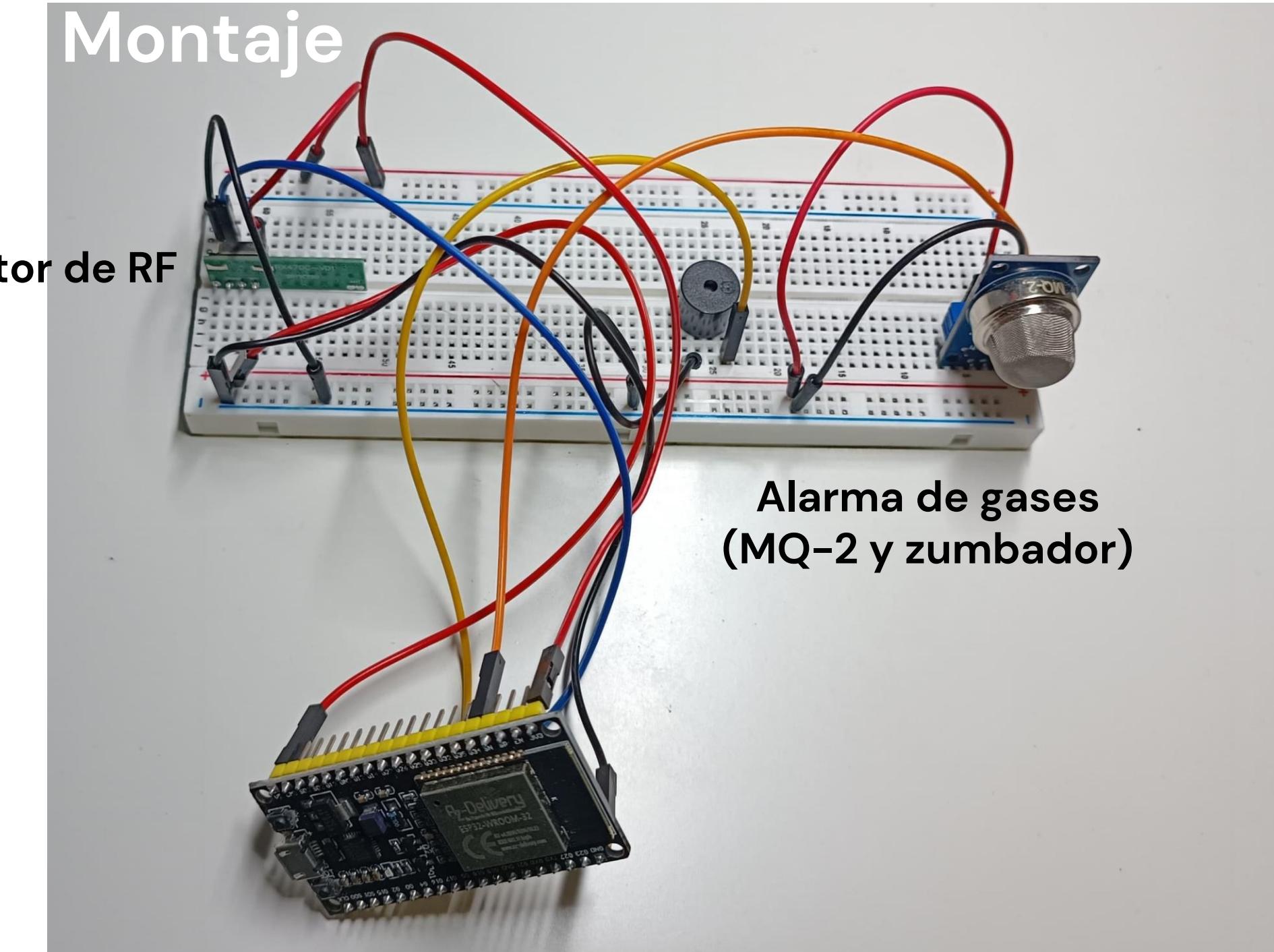


Fuente de alimentación y tierra

Enviar datos Arduino Cloud

### Montaje

#### Receptor de RF

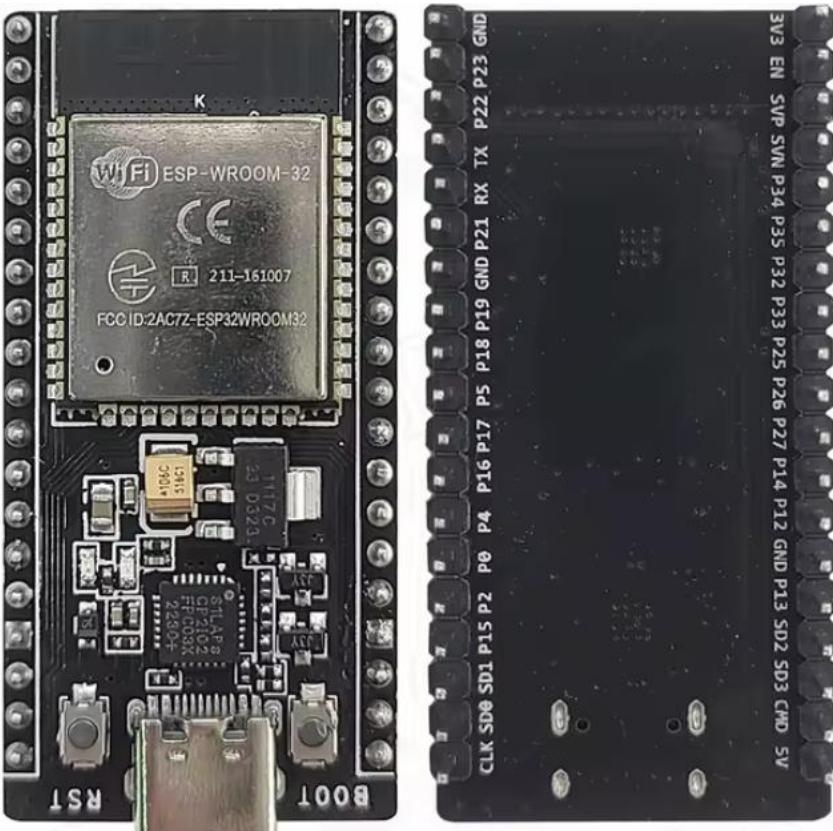


Alarma de gases  
(MQ-2 y zumbador)

# Microcontrolador

## ESP32 SECUNDARIO

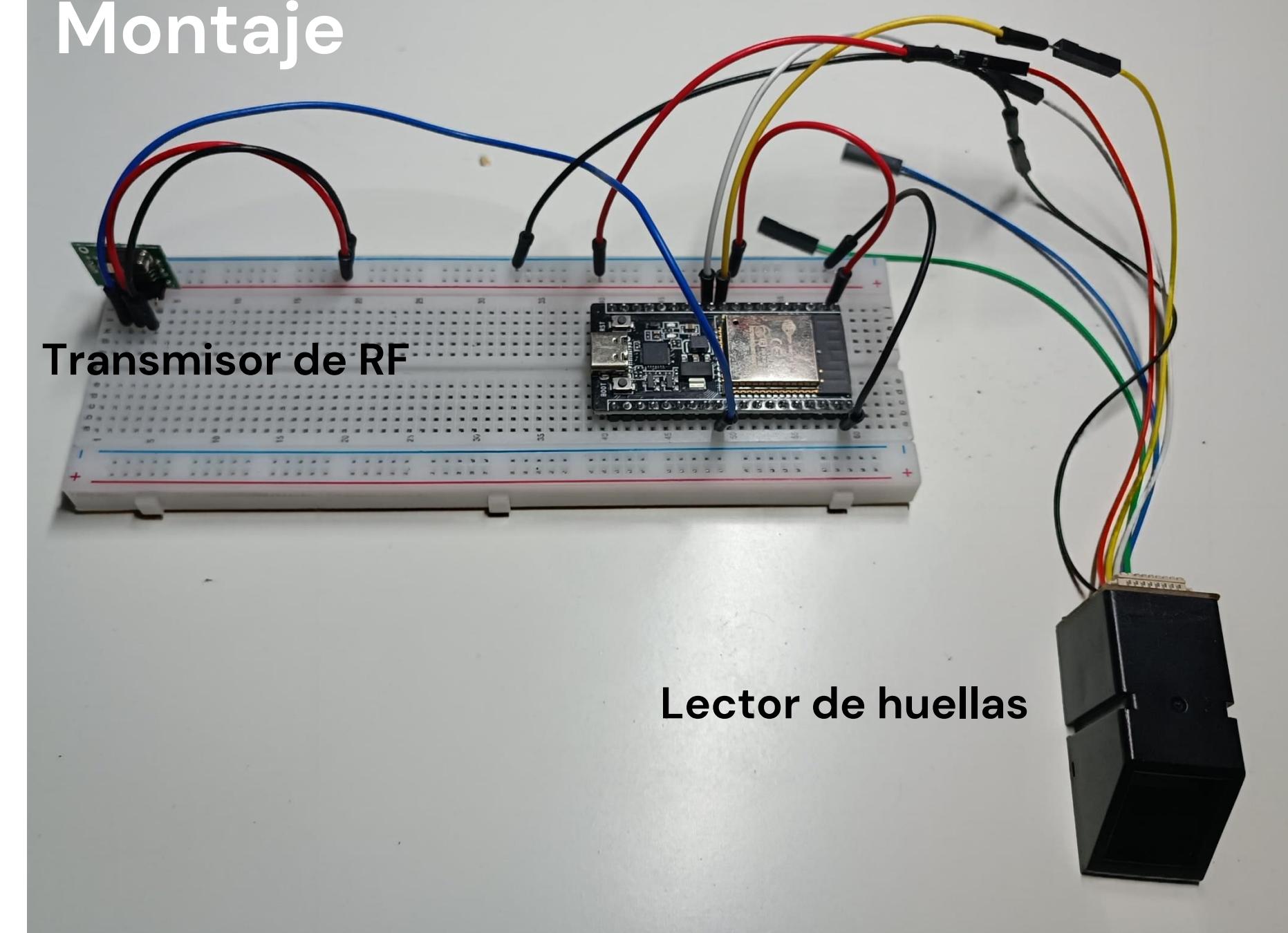
### ESP32 Secundario



- Manejar y sincronizar el transmisor y el lector de huellas
- Fuente de alimentación y tierra

### Montaje

#### Transmisor de RF



# Arduino Cloud

The screenshot shows the Arduino Cloud interface. On the left is a sidebar with options: Sketches, Devices, Things, Dashboards, Triggers, Templates (NEW), Resources, and Courses (All Systems Operational). The main area displays a list of recent files:

All items	Name	Owner	Last modified	Creation date
Trigger	Sensor puertas	Javier Piay	15 jun 2024, 21:26	-
Dashboard	TFM	Javier Piay	15 jun 2024, 21:06	-
Thing	TFM	Javier Piay	13 ago 2024, 15:43	28 may 2024,
Trigger	Camara	Javier Piay	15 jun 2024, 19:48	-
Thing	Proyecto IOT_VIU	Javier Piay	28 may 2024, 13:24	26 feb 2024, 1

**Things** → Creación de las variables  
Sincronización ESP32 principal

**Sketches** → Redacción del código

## CONFIGURACIÓN

Gestión dispositivos IoT en la nube

- **Activar\_alarma**: estado alarma puertas/ventanas
- **Concentración\_gas**: gases sensor MQ-2
- **Face\_detected**: información del reconocimiento facial
- **Fuego**: se activa si concentración de gas > umbral
- **Puerta\_princip**: estado del sensor magnético
- **Huella**: estado lector dactilar

The screenshot shows the configuration details for a TFM device. It includes sections for Cloud Variables, Associated Device, and Network.

**Cloud Variables**

Name	Last Value	Last Update
activar_alarma bool activar_alarma;	false	26 Aug 2024 13:31:34
concentracion_gas int concentracion_gas;	312	26 Aug 2024 15:15:36
face_detected String face_detected;	Javi	26 Aug 2024 13:19:25
fuego bool fuego;	false	26 Aug 2024 13:20:17
huella bool huella;	true	26 Aug 2024 13:31:34
puerta_princip bool puerta_princip;	true	26 Aug 2024 13:19:25

**Associated Device**

ID	Type	Status
e9e83aa0-4a78-4be9-9e	NodeMCU-32S	Online

**Network**

Wi-Fi Name:	LowiDB90
Password:	.....
Secret Key:	.....

8

# Arduino Cloud

## MONITORIZACIÓN



### Triggers



### Dashboard

↓  
Estado variables en tiempo real

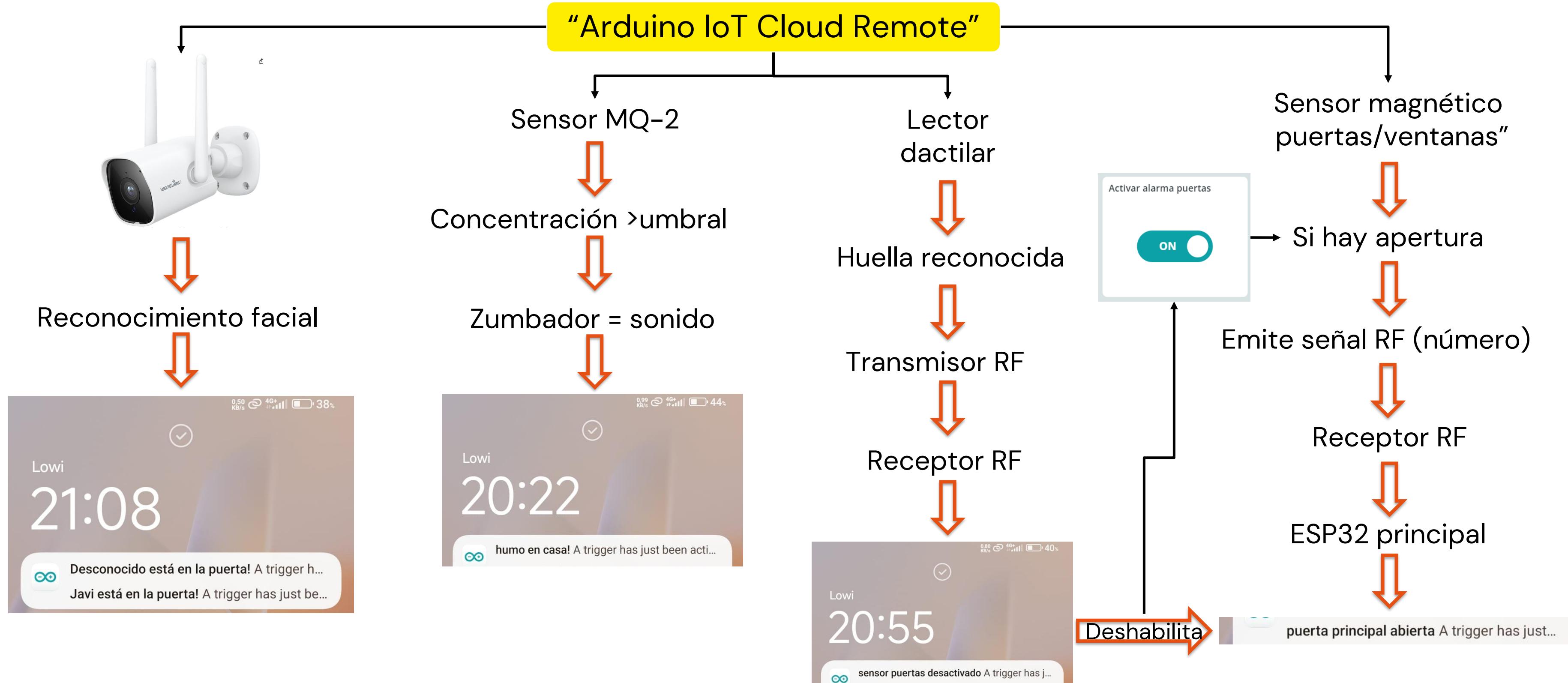
The screenshot shows the "Trigger" configuration interface with the following steps:

- If:** Cloud Variable `face_detected` from TFM is Not empty.
- Then:** Send Push notification to user `javi_piay` with title `{variable.value} está en la puerta!`.
- Actions:** Options to "Send email" or "Send Push notification".

# Sistema

9

## FUNCIONAMIENTO PRÁCTICO



# Resultados

10

## RECONOCIMIENTO FACIAL



```
Procesando carpeta número 2. Nombre:Javi
1/1 ━━━━━━ 0s 144ms/step
1/1 ━━━━━━ 0s 132ms/step
1/1 ━━━━━━ 0s 98ms/step
1/1 ━━━━━━ 0s 101ms/step
1/1 ━━━━━━ 0s 88ms/step
1/1 ━━━━━━ 0s 103ms/step
1/1 ━━━━━━ 0s 117ms/step
1/1 ━━━━━━ 0s 133ms/step
1/1 ━━━━━━ 0s 94ms/step
```

```
1/1 ━━━━━━ 0s 80ms/step
1/1 ━━━━━━ 0s 82ms/step
1/1 ━━━━━━ 0s 86ms/step
1/1 ━━━━━━ 0s 83ms/step
1/1 ━━━━━━ 0s 89ms/step
1/1 ━━━━━━ 0s 82ms/step
1/1 ━━━━━━ 0s 84ms/step
1/1 ━━━━━━ 0s 81ms/step
1/1 ━━━━━━ 0s 90ms/step
1/1 ━━━━━━ 0s 86ms/step
```

# Resultados

10

## ALCANCE RECONOCIMIENTO FACIAL

Penumbra → Rasgos distorsionados → ↑ Dificultad

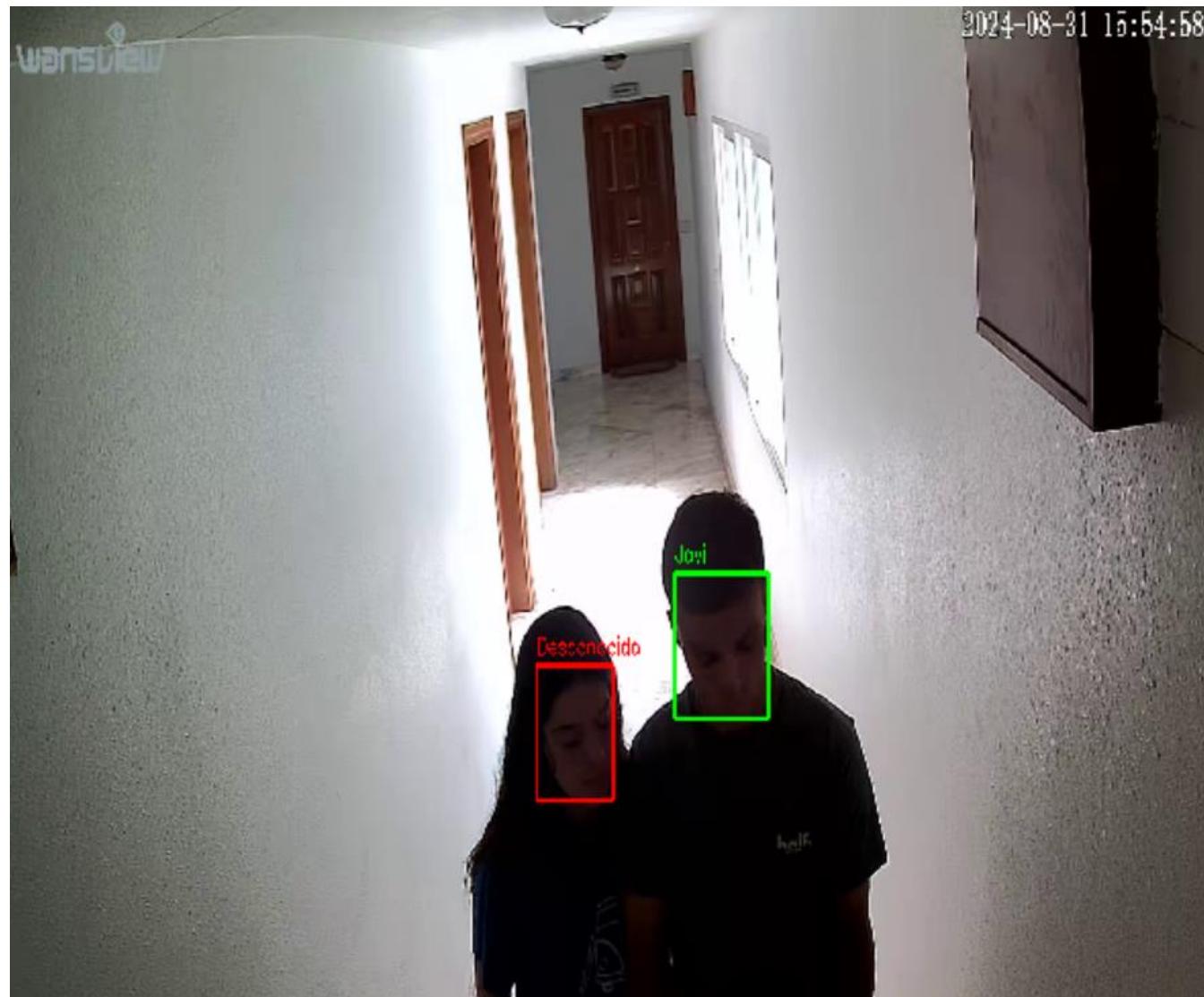


10

# Resultados

## RECONOCIMIENTO SIMULTÁNEO

Penumbra



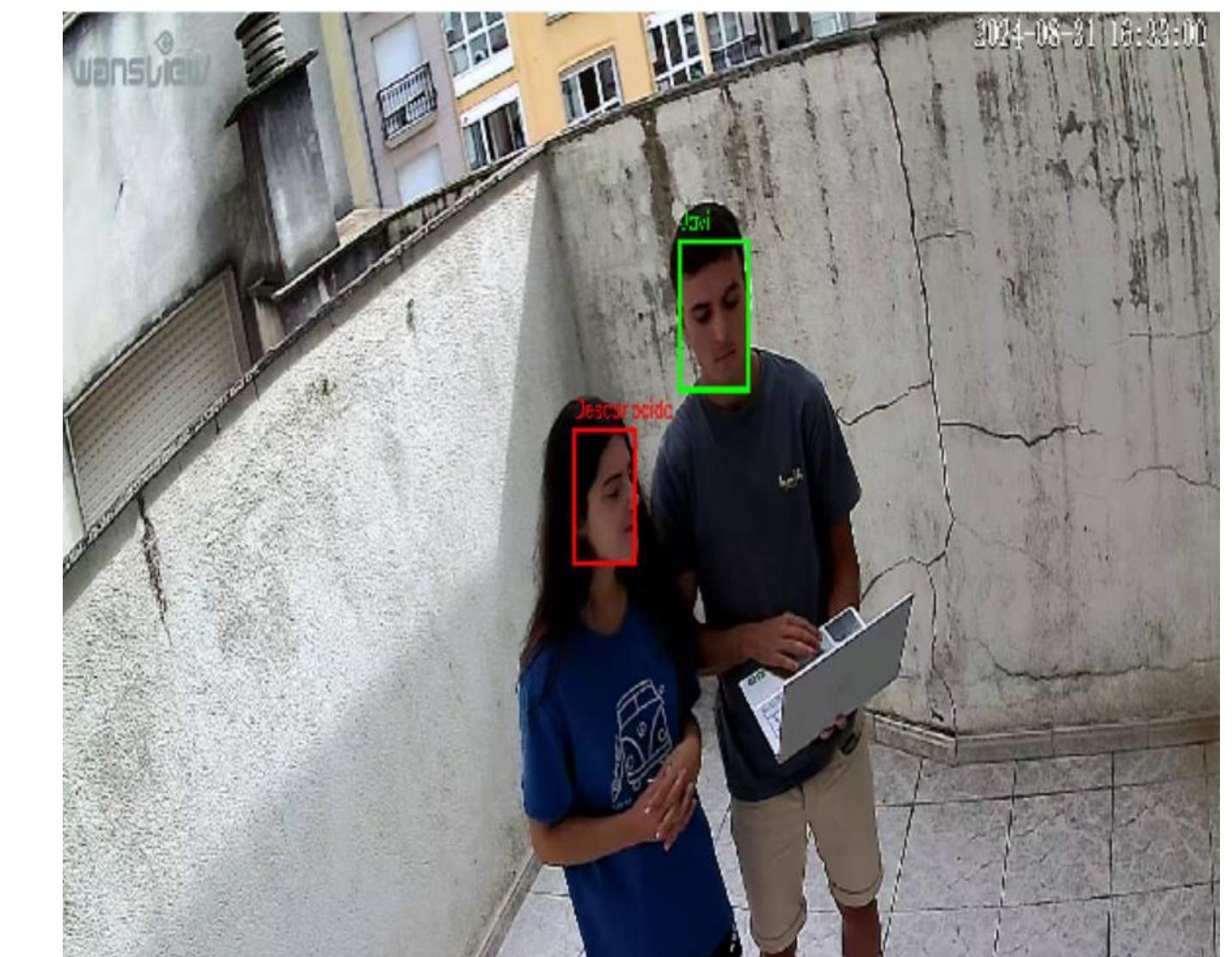
Mismo alcance

Envío de datos



Nombre persona  
reconocida

Buena iluminación



# Resultados

10



## VISIÓN NOCTURNA

### VISIÓN NOCTURNA

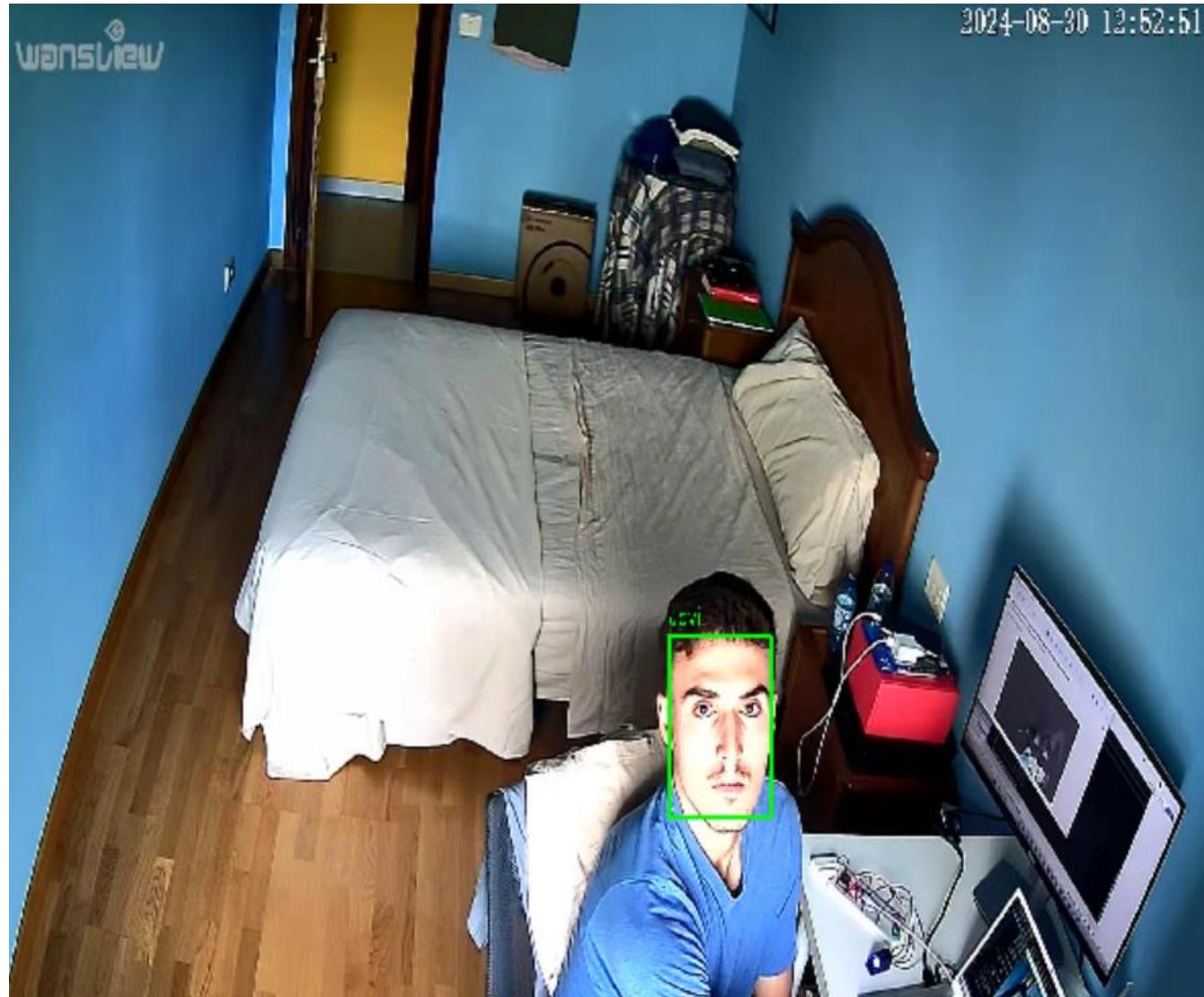
Alcance reducido	1 metro
Falsos negativos	+ 5 metros
Resolución	> penumbra

10

# Resultados

**RENDIMIENTO**

## Grabación fluida



**MAYOR PROBLEMA**

CPU ➔ ≈ 100%

**Solución**

Código



Evitar capturar  
todos los frames

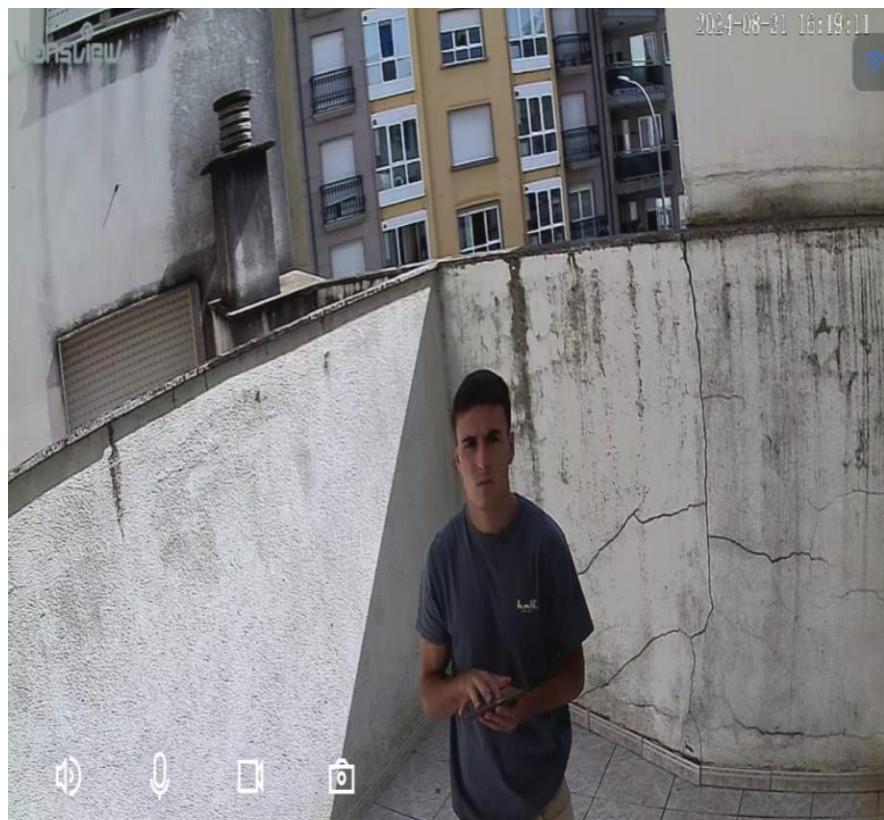
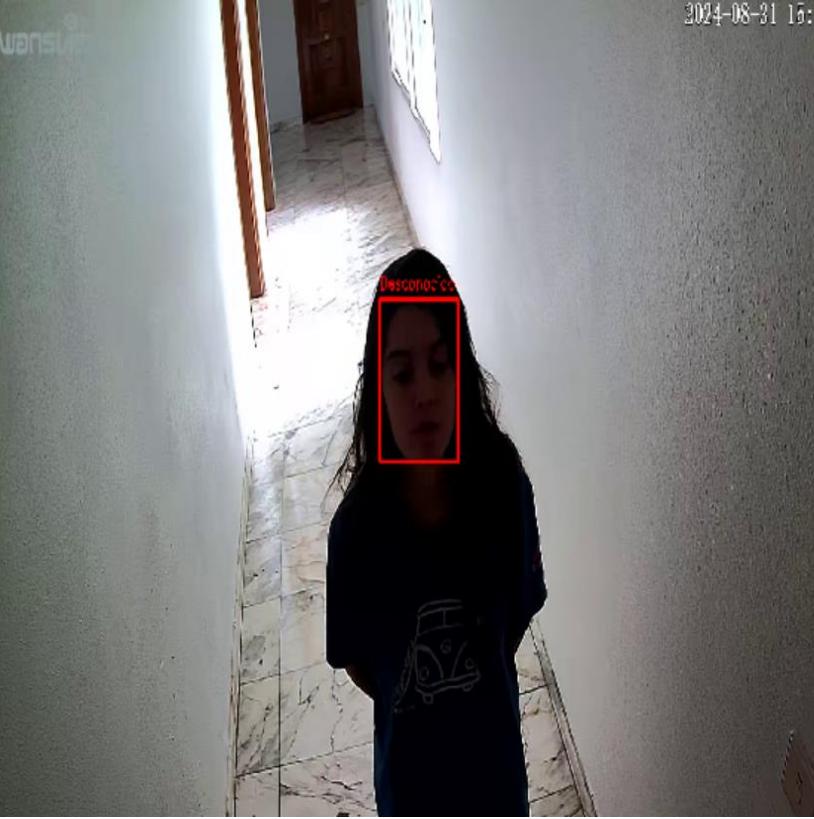
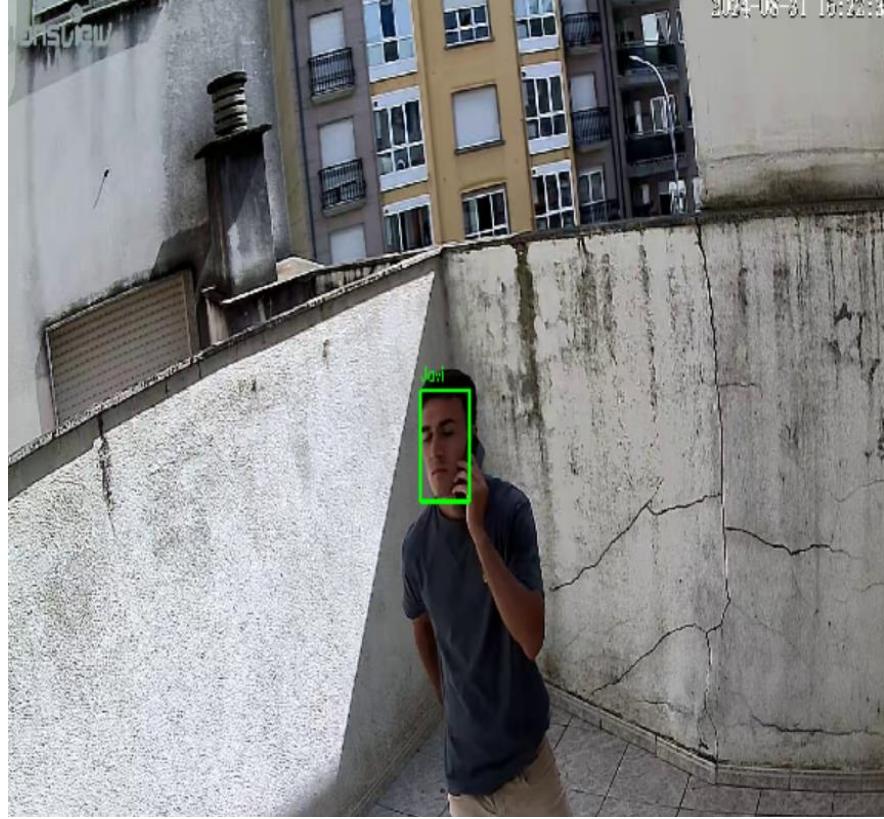
## Grabación congelada



# Resultados

10

## CÁMARA DE VIDEOVIGILANCIA



WANSVIEW Y1

RTSP

App simple

Visión nocturna

NO película de granos

10

# Resultados

## SENSORES Y NOTIFICACIONES

### Funcionamiento correcto

#### Sensores



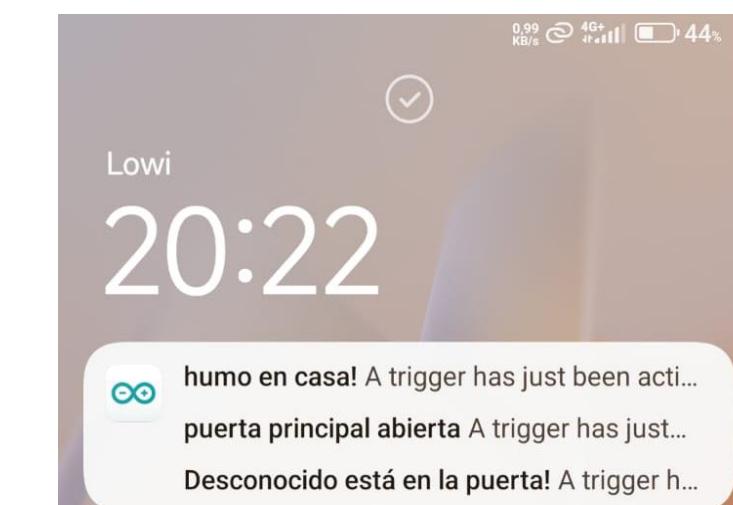
MQ-2 ➔ 2-3 min para calentar

Lector dactilar ➔ Fallos por suciedad

Magnético puertas ➔ Redundancia señales de RF

### Solución

Código evitar lecturas duplicadas

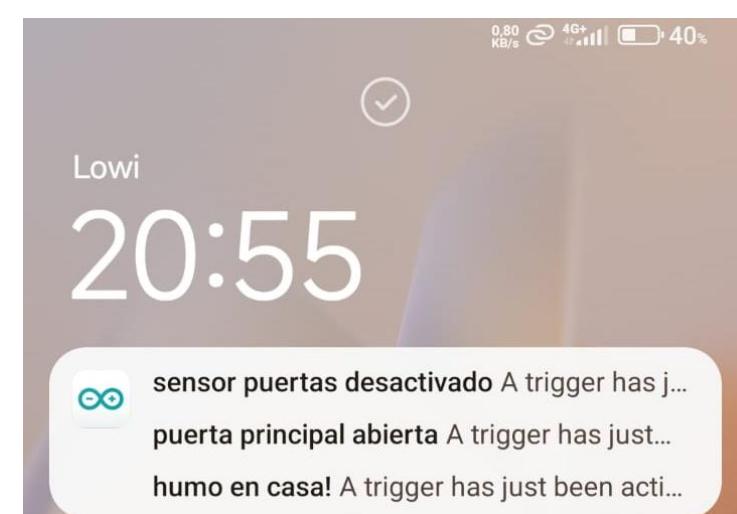
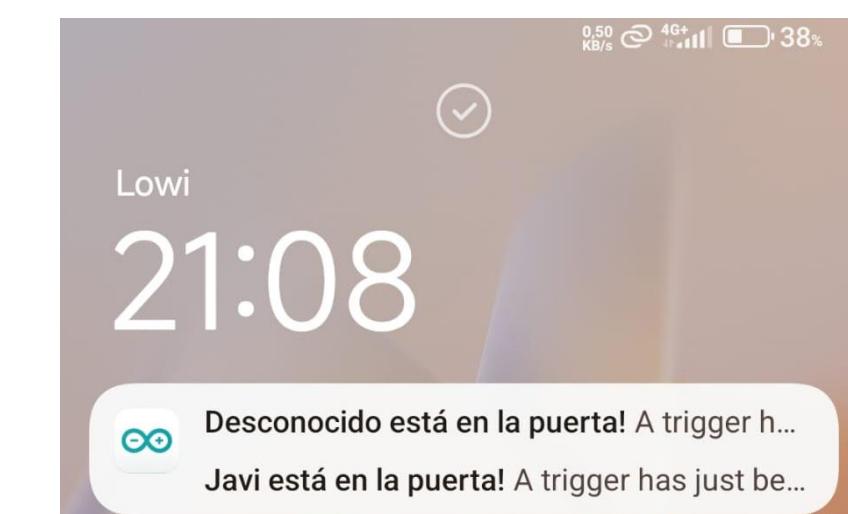


#### Notificaciones



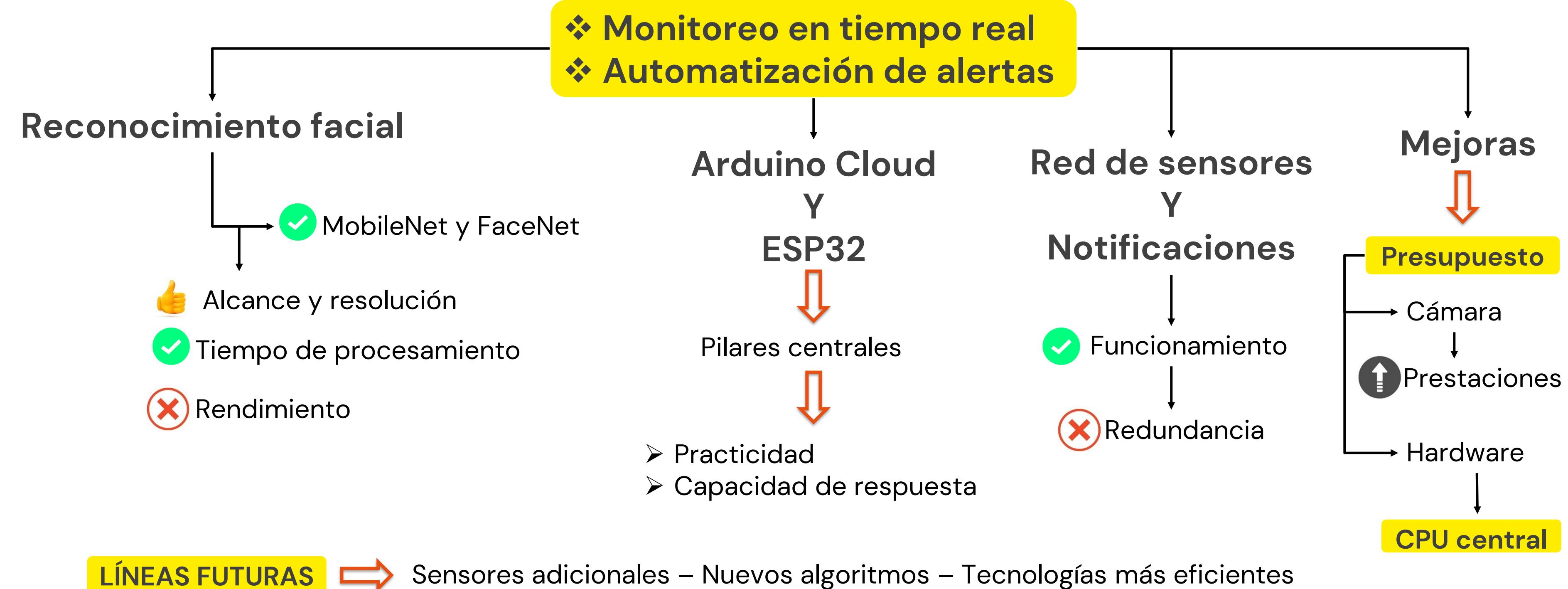
Notificaciones erróneas en el encendido

### Posibles notificaciones



# Conclusiones

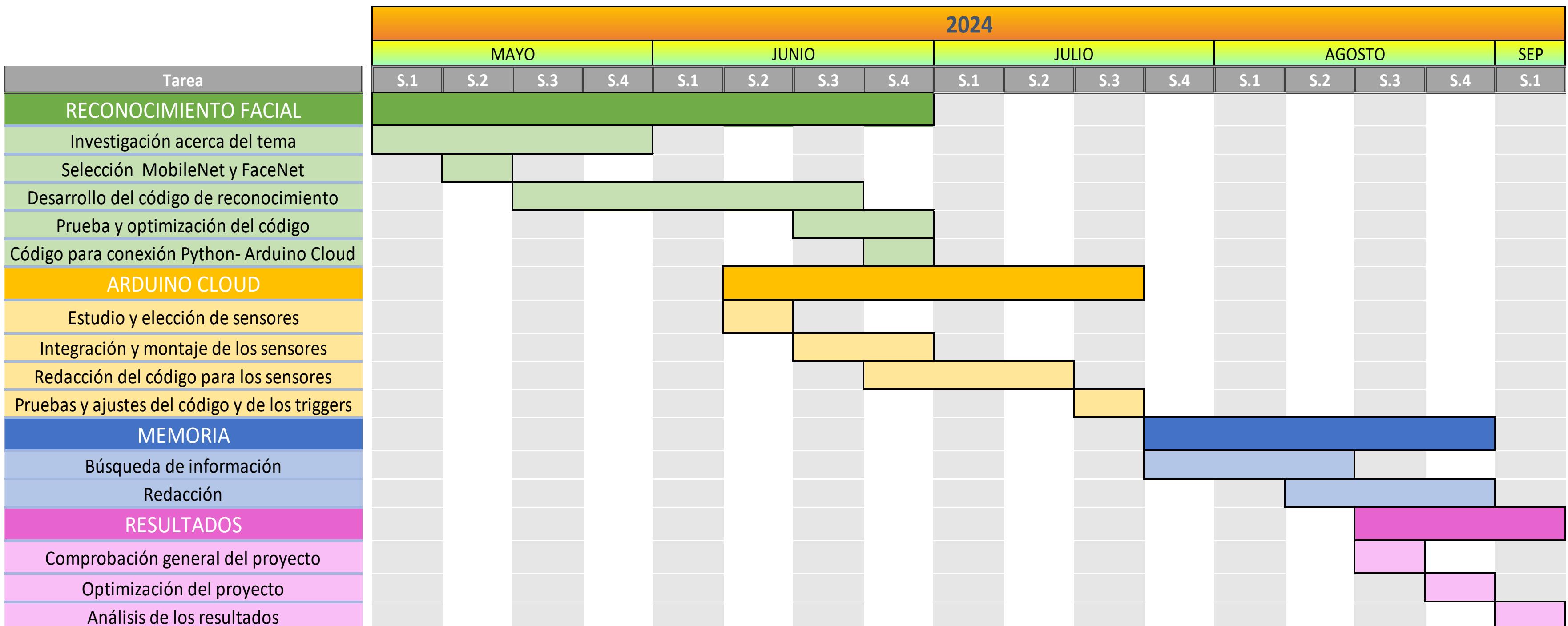
## SISTEMA DE SEGURIDAD



12

# Cronograma

DURACIÓN TOTAL  
4 meses 1 semana



12

# Presupuesto

- ❖ Duración
- ❖ Licencias por 1 año

**COSTE TOTAL**  
**938,97€**

Tabla 7. Coste total del hardware y el software.

	Precio
Hardware	886,68 €
Software	147,65 €
	<b>1034,33 €</b>

viu

Máster en Industria 4.0  
Universidad Internacional de Valencia



Gracias  
por su  
atencion!



Javier Piay Rodríguez