

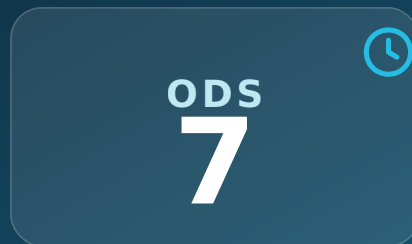


Libro Ares · borrador de contenido y estructura. Las imágenes son **genéricas/de referencia** (ilustraciones y marcos rotulados): reemplazar por fotos y renders del producto real.

ODS 7 · Energía asequible y no contaminante

Casa que ahorra

Línea Ares · Innovadores · 9°



Una herramienta del estudiante · prototipa en papel, construye en MDF

ESTE LIBRO PERTENECE A

NOMBRE DEL ESTUDIANTE

GRADO Y GRUPO

.....

.....

COLEGIO

DOCENTE

.....

.....

Índice

1. Energía en casa

2. La pregunta
3. Lo que vamos a construir
4. El mapa del libro
Antes de empezar · tus comodines
MP1 · Conoce tu ESP32 y conéctalo
MP2 · Enciende a distancia
MP3 · ¿Cuánta energía gasto?
MP4 · La casa inteligente
Del papel al MDF
Lo mostramos
Mi nota
Anexo recortable
Palabras nuevas

La historia

INICIO Érase una vez... **Andrés** vive en una casa donde siempre hay aparatos encendidos.

NUDO La factura de la luz no para de subir y nadie sabe en qué se va la energía.

DESENLACE

Andrés decide investigarlo con tecnología. Vas a construir una **casa inteligente** que se controla por WiFi y muestra cuánta energía gasta, para ahorrar de verdad.

Y aquí empiezas tú: este libro es el camino para que esa historia termine bien. ¡Manos a la obra!

1 · Energía en casa

Gran parte de la energía que gastamos en casa se desperdicia: luces encendidas sin nadie, equipos prendidos de más. El **ODS 7** busca energía limpia y un uso eficiente para todos.

Una casa **inteligente** puede controlarse a distancia y ayudarte a **ahorrar**.



Imagen de referencia: hogar con control inteligente de energía

2 · La pregunta que nos mueve

PREGUNTA QUE DETONA

¿Cómo puede una vivienda ahorrar energía y controlarse a distancia?

Vas a construir una maqueta de casa que controlas desde el celular y que te muestra cuánto consume.

3 · Lo que vamos a construir

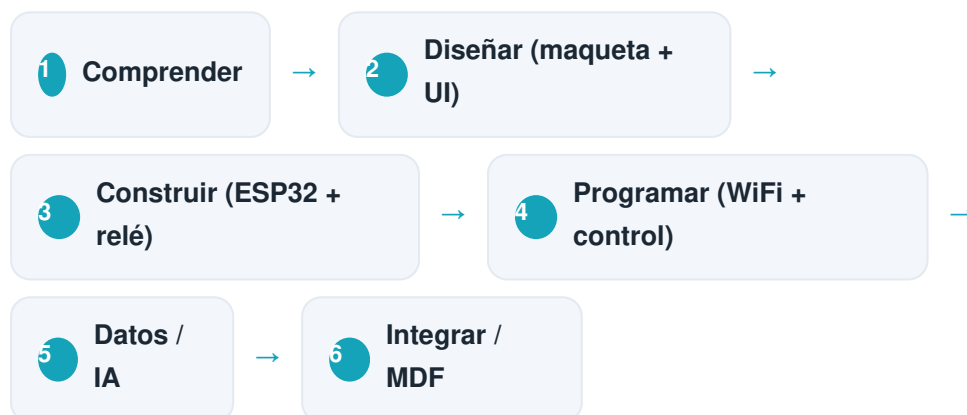
Tu reto es una **casa inteligente (domótica)** con ESP32: controlas sus luces por WiFi y ves su consumo. Papel y luego **MDF**.



Render de referencia: maqueta de casa inteligente en MDF

4 · El mapa del libro

Última banda: nueva placa con internet (**ESP32**) y tu primera **app/ dashboard**. Vamos así:



Antes de empezar · tus comodines

Tu placa nueva es la **ESP32**: como la Pico, pero con **WiFi**. Eso permite controlar y ver tu casa **a distancia** (domótica).

COM-PL-ESP

ESP32 (con WiFi) **NUEVO**

Placa · Cartilla · Academy

COM-BS-WIFI

Conexión WiFi y red **NUEVO**

Base · Cartilla · Academy

COM-AC-RELE

Relé (controla la luz) **REPASO**

Actuador · Cartilla · Academy

¿Qué hace 'inteligente' a una casa?

Una casa domótica se controla y se mide **a distancia**, por WiFi. Para eso el programa se **conecta a la red**, **responde a órdenes** y **protege** el sistema si la conexión falla.

Red / WiFi = la ESP32 se conecta a internet y puede recibir órdenes y enviar datos.

Manejo de errores = try / except: si la conexión falla, el programa no se cae, avisa y sigue.

Función handler = el código que se ejecuta cuando llega una orden (encender/apagar).

Verifica que entendiste: ¿Qué permite el WiFi en tu casa? ¿Qué pasa si la conexión se cae?

Pensamiento computacional primero: haz la **actividad desenchufada** de apertura (sin computador) y practica la misma lógica del proyecto en **Python (retos)** antes de construir. Cada microproyecto se diseña con su **diagrama de flujo** o **pseudocódigo**.

1

Conoce tu ESP32 y conéctalo

Microproyecto 1 · primera conexión WiFi

Entrar a la red

Lo primero de la domótica: que la ESP32 **se conecte al WiFi**. Sin red, no hay control a distancia.

OBJETIVOS

- Conectar la ESP32 al WiFi.
- Confirmar la conexión con su dirección IP.

MATERIALES

DEL ANEXO RECORTABLE

- Plantilla **D-1** (base de la placa)

COMPONENTES

- ESP32
- Cable USB
- Red WiFi (datos del docente)

CONSTRUCCIÓN PASO A PASO

1



Escribe el **nombre (SSID)** y la **clave** del WiFi del colegio (pídelos al docente). *Seguridad:* no compartas la clave fuera de clase.

Por qué: El ESP32 necesita identificarse en la red con su nombre (SSID) y clave, como tú con tu usuario.

2



Conéctate con `wlan.connect(ssid, clave)` y espera.

Por qué: Conectarse es pedir entrada a la red; por eso hay que **esperar** a que responda antes de seguir.

Ojo: si nunca conecta, revisa que el nombre y la clave estén **exactos** (mayúsculas incluidas) y que sea red de 2.4 GHz.

Compruébalo: Imprime `wlan.isconnected()`: ¿da True?

3



imprime `wlan.ifconfig()`. ¿Aparece una **dirección IP**? Esa es la 'dirección' de tu casa en la red.



Foto: ESP32 conectada por USB mostrando su IP en consola

⇒ Diseña tu algoritmo (antes de programar)

Antes de escribir el programa, **dibuja el diagrama de flujo** o escribe el **pseudocódigo** de lo que hará tu proyecto. El código es la consecuencia de pensar el algoritmo.



Pseudocódigo:

PROGRAMACIÓN (MICROPYTHON (ESP32))

```
import network, time
wlan = network.WLAN(network.STA_IF)
wlan.active(True)
wlan.connect('RED_COLEGIO', 'clave')
while not wlan.isconnected():
    time.sleep(1)
print('Conectado. IP:', wlan.ifconfig()[0])
```

DISEÑO

Diseña la base de la placa. Criterio: protegida y con el cable accesible.
Bocétala.

RESPONDE EN TU LIBRO

¿Qué es la dirección IP de tu ESP32?

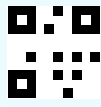
¿Por qué la red debe ser de 2.4 GHz?

EVALUACIÓN DEL MICROPROYECTO (LA LLENA EL DOCENTE)

Criterio	1	2	3	4	Puntos
Construcción	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Programación / lógica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Diseño	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Preguntas del libro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Trabajo y proceso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
TOTAL					___/20

.....
Nota

.....
Firma del docente



Jugar en Academy

Actividad del microproyecto 1 · Conoce tu ESP32 y conéctalo

Actividad **táctil** para repasar jugando (el docente lee en voz alta).

Escanea el QR o [ábrela aquí ▶](#).

2

Enciende a distancia

Microproyecto 2 · controlar la luz por WiFi

Recibir una orden y actuar

La ESP32 abre un **pequeño servidor**: cuando llega la orden 'encender', una **función** prende la luz por el relé. Y la protegemos con **try/except**.

OBJETIVOS

- Encender/apagar un LED o relé por WiFi.
- Manejar el caso en que algo falle.

MATERIALES

DEL ANEXO RECORTABLE

- Plantilla **D-2** (lámpara)

COMPONENTES

- ESP32
- Relé + LED/lámpara

CONSTRUCCIÓN PASO A PASO

1



Conecta el **relé** a un pin de salida (el relé enciende la luz; la ESP32 sola no aguanta una lámpara).

Por qué: El **relé** es un interruptor que la ESP32 controla: deja que una señal pequeña encienda una lámpara grande sin dañar la placa.

2



Programa: si llega /on → `rele.on()`; si /off → `rele.off()`. Envuélvelo en **try/except**.

Por qué: El `try/except` evita que un mensaje raro tumba el programa: si algo falla, lo atrapa y sigue.

Ojo: sin `try/except`, un dato raro de la red **tumba** el programa.

3



desde el navegador del celular escribe la IP y /on. ¿Se enciende la luz?



Foto: lámpara encendida desde el celular por WiFi

⇒ Diseña tu algoritmo (antes de programar)

Antes de escribir el programa, **dibuja el diagrama de flujo** o escribe el **pseudocódigo** de lo que hará tu proyecto. El código es la consecuencia de pensar el algoritmo.



Pseudocódigo:

```
def manejar(orden):
    try:
        if orden == '/on':
            rele.on()
        elif orden == '/off':
            rele.off()
    except Exception as e:
        print('Fallo, sigo:', e)
```

DISEÑO

Diseña la lámpara. Criterio: segura (cables aislados) y fácil de ver al encender.

RESPONDE EN TU LIBRO

¿Qué hace la función manejar()?

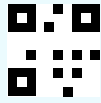
¿Para qué sirve el try/except aquí?

EVALUACIÓN DEL MICROPROYECTO (LA LLENA EL DOCENTE)

Criterio	1	2	3	4	Puntos
Construcción	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Programación / lógica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Diseño	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Preguntas del libro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Trabajo y proceso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
TOTAL					___/20

.....
Nota

.....
Firma del docente



Jugar en Academy

Actividad del microproyecto 2 · Enciende a distancia

Actividad **táctil** para repasar jugando (el docente lee en voz alta).
Escanea el QR o [ábrela aquí ▶](#).

3

¿Cuánta energía gasto?

Microproyecto 3 · medir para ahorrar

Medir para decidir

Para ahorrar, primero hay que **medir**. Guardamos el consumo en una **estructura de datos** y calculamos cuánto se gasta.

OBJETIVOS

- Medir (o simular) el consumo.
- Guardar y resumir los datos.

MATERIALES

DEL ANEXO RECORTABLE

- —

COMPONENTES

- ESP32
- Sensor de corriente (o simulación)

CONSTRUCCIÓN PASO A PASO

1



Lee el **consumo** (sensor de corriente o un valor simulado) cada cierto tiempo.

Por qué: Medir cada cierto tiempo crea una **serie** de datos: con un solo dato no sabes cuándo gastas más.

2



Guarda cada lectura con su hora en una **lista de diccionarios**.

Por qué: Guardar lectura + hora juntas te deja responder luego **cuándo** pasó, no solo cuánto.

Ojo: guardar solo el último valor — entonces no puedes ver el **total** ni el **pico**.

3



calcula el consumo total y la hora de mayor gasto.
¿Coincide con cuando hay más luces prendidas?



Foto: consola mostrando consumo por hora

⇒ Diseña tu algoritmo (antes de programar)

Antes de escribir el programa, **dibuja el diagrama de flujo** o escribe el **pseudocódigo** de lo que hará tu proyecto. El código es la consecuencia de pensar el algoritmo.



Pseudocódigo:

PROGRAMACIÓN (PYTHON / MICROPYTHON)

```
registro = []  
lectura = {'hora': 14, 'watts': 120}
```

```
registro.append(lectura)
total = sum(r['watts'] for r in registro)
pico = max(registro, key=lambda r: r['watts'])
print('Total:', total, '| Pico a las', pico['hora'])
```

DISEÑO

Diseña cómo mostrar el gasto. Criterio: claro para decidir **qué apagar**.

¿Gráfico, semáforo de colores?

📖 RESPONDE EN TU LIBRO

¿Por qué guardas la hora junto al consumo?

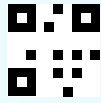
¿A qué hora gasta más tu casa?

EVALUACIÓN DEL MICROPROYECTO (LA LLENA EL DOCENTE)

Criterio	1	2	3	4	Puntos
Construcción	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Programación / lógica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Diseño	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Preguntas del libro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Trabajo y proceso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
TOTAL					___/20

.....
Nota

.....
Firma del docente



Jugar en Academy

Actividad del microproyecto 3 · ¿Cuánta energía gasto?

Actividad **táctil** para repasar jugando (el docente lee en voz alta).
Escanea el QR o [ábrela aquí ▶](#).

4

La casa inteligente

Microproyecto 4 · integrar y dar forma

Todo junto: una casa que ahorra

Unes el WiFi, el control remoto y la medición en una **maqueta de casa domótica** que se controla y se mide a distancia (ODS 7: usar la energía mejor).

OBJETIVOS

- Integrar control + medición.
- Probar el ahorro real.

MATERIALES

DEL ANEXO RECORTABLE

- Plantilla **D-4** (casa)

COMPONENTES

- Todo lo anterior
- Maqueta D-4

CONSTRUCCIÓN PASO A PASO

1



Monta la casa D-4 con sus luces controladas y el medidor.
Por qué: La maqueta vuelve real el ahorro: muestra dónde se controla la luz y dónde se mide.

2



Corre el sistema: controla luces por WiFi y registra el consumo.

Por qué: Convergen los tres hilos: controlar por WiFi (programa), el relé y las luces (circuito) y la casa (diseño).

Ojo: si se cae al perder WiFi, te faltó proteger con try/except y reconectar.

3



apaga luces desde el celular y mira bajar el consumo.
¿Demuestra el ahorro?



Foto: maqueta de casa inteligente terminada

⇒ Diseña tu algoritmo (antes de programar)

Antes de escribir el programa, **dibuja el diagrama de flujo** o escribe el **pseudocódigo** de lo que hará tu proyecto. El código es la consecuencia de pensar el algoritmo.



Pseudocódigo:

PROGRAMACIÓN (MICROPYTHON (ESP32))

```
# ciclo principal (resumen)
while True:
    orden = revisar_red()      # ¿llegó una orden?
```

```
manejar(orden)           # actúa (con try/except)
registrar_consumo()      # mide y guarda
time.sleep(1)
```

DISEÑO

Da forma a tu casa. Criterio: que se vean las luces y el medidor. Decora y nombra.

IA · CONVERSA

Pide a la IA: «¿qué hábitos y aparatos gastan más energía en una casa?» y proponle 3 mejoras a la tuya.

📖 RESPONDE EN TU LIBRO

¿Cómo demuestra tu casa que ahorra?

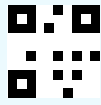
¿Qué pasa si se cae el WiFi y cómo lo resolviste?

EVALUACIÓN DEL MICROPROYECTO (LA LLENA EL DOCENTE)

Criterio	1	2	3	4	Puntos
Construcción	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Programación / lógica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Diseño	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Preguntas del libro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Trabajo y proceso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
TOTAL					___/20

.....
Nota

.....
Firma del docente



Jugar en Academy

Actividad del microproyecto

4 · La casa inteligente

Actividad táctil para repasar jugando (el docente lee en voz alta). Escanea el QR o [ábrela aquí](#) ▶.

Del papel al MDF · el prototipo final

Ahora construyes tu casa en **madera (MDF)**: una maqueta firme con sus cuartos, luces y el medidor a la vista.



Render de referencia: casa inteligente en MDF

Lo mostramos

Controla tu casa frente a la clase y muestra el ahorro. Vuelve a la pregunta: ¿logra ahorrar energía?

¿Qué mejorarías si lo hicieras otra vez?

Mi nota · rúbrica final del proyecto

El docente evalúa el **prototipo final**. La **convergencia** (que los tres hilos funcionen juntos) vale el doble.

Criterio	1	2	3	4	Puntos
Programación	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Robotización	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Diseño	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
Convergencia (x2) — prueba de "quita un hilo"	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/8
Proceso y comunicación	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	___/4
TOTAL					___/24

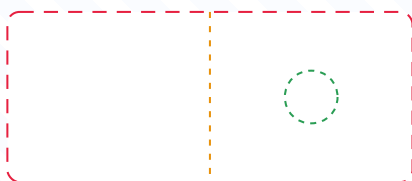
.....
Nota final

.....
Firma del docente

Anexo recortable

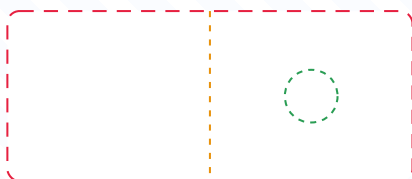
✂ Recorta por las líneas rojas. --- Dobla por las punteadas. ● Fija el componente en los puntos marcados. **No recortes las páginas de guía ni de evaluación.**

Plantilla C9-2 · lámpara



conecta aquí el relé ●

Plantilla C9-4 · maqueta de casa



dobra por ---

Palabras nuevas

ESP32

Placa con WiFi para domótica e IoT.

WiFi / red

Conexión que permite controlar y ver a distancia.

try / except

Atrapa errores para que el programa no se caiga.

Función handler

Código que responde cuando llega una orden.

Relé

Interruptor que deja que la ESP32 encienda una lámpara.