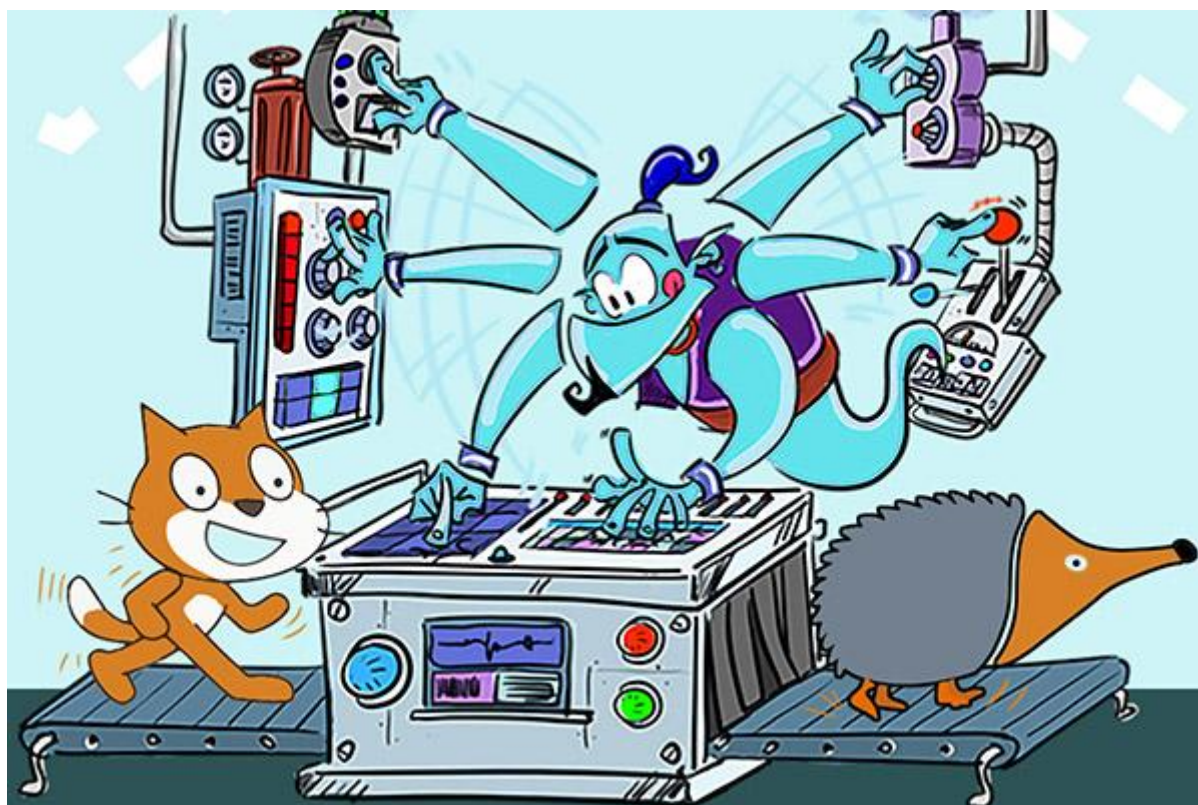


# 5. Inteligencia Artificial

- [Qué es la IA](#)
- [Tipos IA](#)
- [MONTAJE 13 Reconocimiento de texto](#)
- [MONTAJE 14 Reconocimiento de imágenes](#)
- [MONTAJE 14BIS RECONOCIMIENTO DE MATRÍCULAS](#)

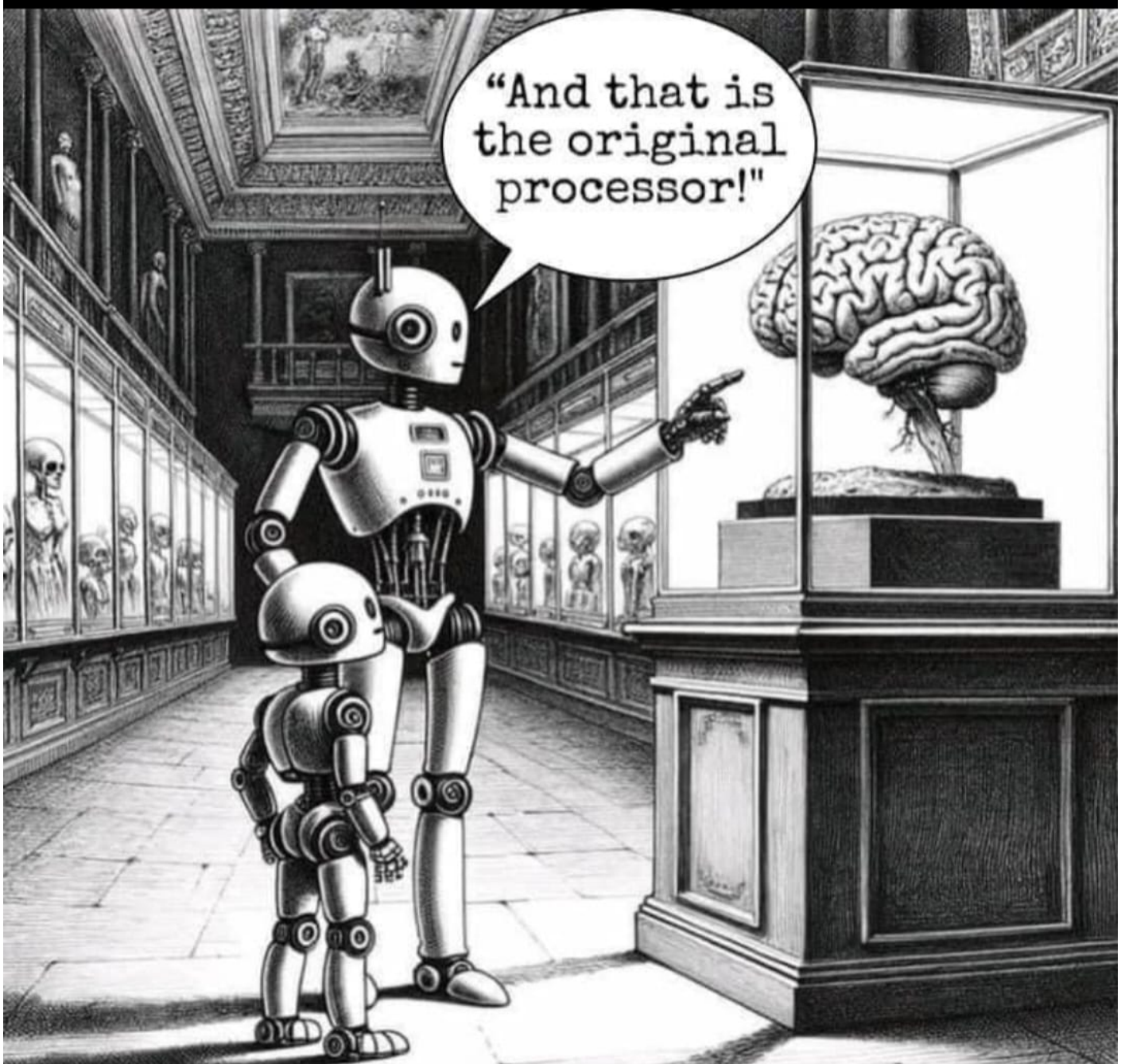
# Qué es la IA

Con EchidnaScratch podemos integrar fácilmente la inteligencia artificial definiendo previamente los modelos en Learning ML



CC-BY-SA Fuente <https://echidna.es/a-programar/echidnaml/>

La **inteligencia artificial** se refiere al conjunto de algoritmos realizan tareas complejas, que son capaces de realizar procedimientos que imitan el funcionamiento de la mente, como son **aprender y tomar decisiones**.



De <https://timoelliott.com/blog/cartoons/artificial-intelligence-cartoons> en <https://timoelliott.com/>

# Tipos IA

**Viendo esta clasificación ¿Cuál es el tipo de IA que se usa en EchidnaScratch ?**

Muy fácil: **ML Aprendizaje automático supervisado**

## ML : APRENDIZAJE AUTOMATICO O MACHINE LEARNING

Aquí los algoritmos aprenden a partir de datos, y ellos buscan patrones para identificar. **Cuanto más datos, mejores resultados.**

El principio es básico. Queremos diferenciar silla mesa: Analiza datos, localiza patrones (cuatro patas, algo común, el tamaño y forma, patrones que diferencian como el respaldo....) con esas características, puede realizar un entrenamiento, y por lo tanto esta preparado para realizar predicciones.

Hay varios tipos de ML:

- **ML Aprendizaje automático supervisado** Donde los datos se les da ya etiquetados
- **ML Aprendizaje automático no supervisado:** Aquí es el propio algoritmo que tiene que encontrar relaciones en el proceso de identificación de patrones y por lo tanto etiquetar categorías.
- **ML Aprendizaje semisupervisado.** El modelo se entrena con datos etiquetados y no etiquetados. Lo normal es primero los etiquetados (pues son los que implican mayor coste y dinero) y si el modelo ya empieza a funcionar bien, ya procesa no etiquetados para su optimización.

## RL : APRENDIZAJE REFORZADO O REINFORCEMENT LEARNING

Es un modelo que busca la optimización. Utiliza resultados anteriores e intenta mejorarlos. Para buscar esto, se definen estados con unos parámetros que implican premio o penalización. La máquina busca soluciones y acciones para buscar los premios y huir de las penalizaciones.

## DL : APRENDIZAJE PROFUNDO O DEEP LEARNING

Es un subconjunto del ML no supervisado, Se crea una red neuronal organizadas en capas, cada capa especializada, por ejemplo la primera de entrada recoge los datos, las intermedias, ya ocultas, identifican patrones y la última, la salida que recoge la información procesada anteriores, la que toma las decisiones.



## PLN : PROCESAMIENTO DE LENGUAJE NATURAL

Utiliza conceptos de DL y ML para comprender el lenguaje humano. Una aplicación son los *chatbots* . Que pueden ser:

- dirigidos, es decir va tomando decisiones por etapas tipo árbol y da respuestas automáticas
- no dirigido: produce respuestas en función del contexto y las decisiones del usuario.
- híbridas que intenta producir una conversación fluida.

## EJEMPLOS



### LEARNING ML

<https://web.learningml.org/en/home/>



### ML: MACHINE LEARNING FOR KIDS

<https://machinelearningforkids.co.uk/#!/welcome>

Ver nuestro tutorial en <https://libros.catedu.es/books/robotica-educativa-con-mbot/page/maquina-educable-con-machine-learning-for-kids-y-scratch>



### DL REDES NEURONALES : QUICKDRAW

<https://quickdraw.withgoogle.com/> Es un videojuego donde se ve claramente como se va perfilando las decisiones en forma de árbol y como la etapa final va reconociendo tu dibujo. En esta red, la primera etapa de la red neuronal son trazos que dibujas, y las neuronas intermedias son fruto de una extensa base de datos de dibujos. La última etapa es la decisión basada en las anteriores neuronas.



PRUEBALO ES SUPERDIVERTIDO, INCLUSO EN UN MÓVIL



### **PLN: CHATGPT COMO FUNCIONA**

<https://www.youtube.com/embed/FdZ8LKijBhQ>

### **DL CREA TU PEQUEÑA RED NEURONAL CON PYTHON**

[https://www.youtube.com/embed/iX\\_on3VxZzk?list=PLZ8REt5zt2Pn0vfjjTAPaDVSACDvnuGiG](https://www.youtube.com/embed/iX_on3VxZzk?list=PLZ8REt5zt2Pn0vfjjTAPaDVSACDvnuGiG)

### **DL: GOOGLE DEEPMIND**

Google deepmind desarrolló una red neuronal para aprender a jugar videojuegos, y ganó a varios campeones humanos en videojuegos complejos (de los que inicialmente la máquina desconocía las reglas). Puedes ver en <https://deepmind.google/> los diferentes proyectos actuales.



# MONTAJE 13 Reconocimiento de texto

Vamos a encender y apagar un objeto de luz usando texto predictivo con IA

Al arrancar EchidnaScratch nos vamos a LEARNING ML, nos vamos a entrenar y que entrene con la clase **APAGA** las palabras que se nos ocurra:

The screenshot shows the EchidnaScratch LearningML interface. At the top, there are three buttons: 'Echidna board connected' (green), 'EchidnaScratch' (orange), and 'LearningML' (green). The 'LearningML' button is highlighted with a yellow box and a red circle with the number 1. Below these buttons is a green header bar with a globe icon, a dropdown menu labeled 'Archivo', a text input field with 'sin nombre', a dropdown menu labeled 'Aprende', and a toggle switch labeled 'Modo avanzado desactivado'. The main area is divided into three columns: '1. Entrenar', '2. Aprender', and '3. Probar'. In the '1. Entrenar' column, there is a button labeled '+ Añadir nueva clase de textos' highlighted with a yellow box and a red circle with the number 2. Below this button is a list of example texts: 'Quiero descansar', 'Me voy a dormir', 'Menos luz', 'Más oscuridad', and 'Apaga'. The text 'Apaga (5)' is highlighted with a yellow box and a red circle with the number 3. In the '2. Aprender' column, there is a dropdown menu labeled 'Lenguaje de los textos' with 'Español' selected, and a button labeled 'Aprender a reconocer textos'. In the '3. Probar' column, there is a text input field labeled 'Expresión' and a button labeled 'Comprobar'. A yellow arrow points from the 'LearningML' button to the 'Añadir nueva clase de textos' button. A red circle with the number 4 is next to a '+' icon at the bottom of the '1. Entrenar' column. A red circle with the number 5 is next to the text 'Menos luz'.

Y con la clase **ENCIENDE** otras tantas palabras

**Enciende (5)**

Se esta haciendo de noche

Quiero leer

No veo bien

Enciende la luz

Enciende la bombilla

+ -

Pulsamos que entrene (tarda un poco) y cuando nos diga que esta listo para usar el modelo, vamos a comprobarlo, como vemos, la palabra **MUCHA LUZ no aparece en las entrenadas**, pero detecta que tiene más posibilidades que lo que queremos es apagar la luz

Archivo ▾ sin nombre

Aprende ▾ ☐ Modo avanzado desactivado

Acerca de

### 1. Entrenar

Primero necesito algunos textos de ejemplo

+ Añadir nueva clase de textos

**Apaga (5)**

Quiero descansar

Me voy a dormir

Menos luz

Más oscuridad

### 2. Aprender

Llegó el momento de aprender a clasificar textos

Lenguaje de los textos Español ▾

**1** Aprender a reconocer textos

### 3. Probar

Introduce términos nuevos y comprueba si se clasifican correctamente

Expresión **2**

Mucha luz

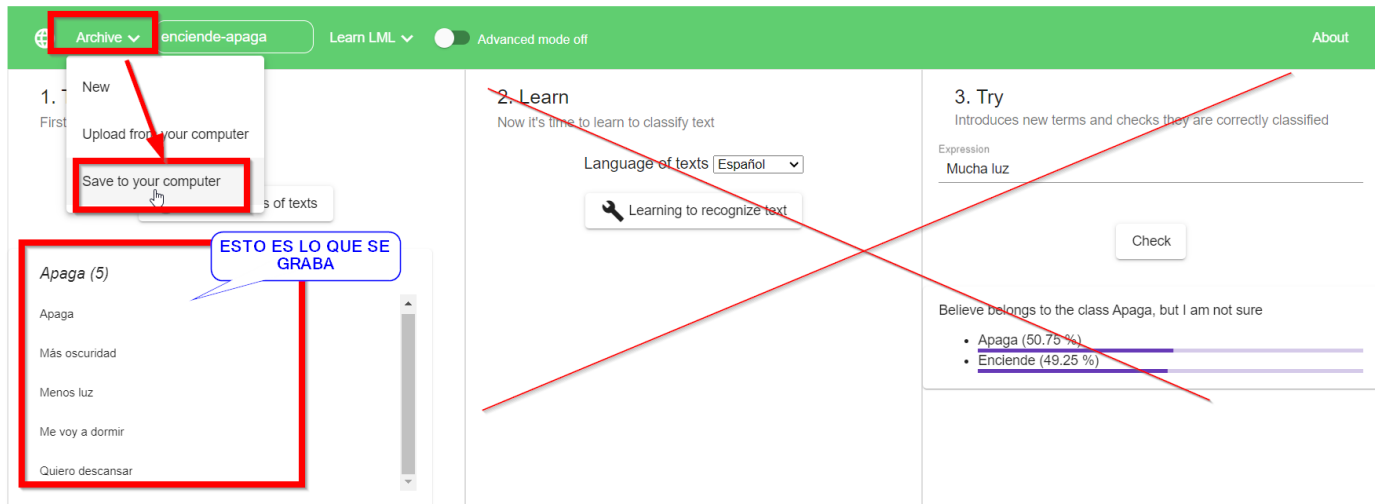
**3** Comprobar

Creo que pertenece a la clase Apaga, aunque no estoy muy segura

- Apaga (53.78 %)
- Enciende (46.22 %)

**Acuerdate de** grabarlo lo voy a grabar como enciende-apaga.json, se graba LOS DATOS DE ENTRENAMIENTO luego al volverlo a abrir, tienes que volver a crear el modelo





Archive ▼ enciende-apaga Learn LML ▼ Advanced mode off About

1. First

New

Upload from your computer

Save to your computer

s of texts

ESTO ES LO QUE SE GRABA

Apaga (5)

Apaga

Más oscuridad

Menos luz

Me voy a dormir

Quiero descansar

2. Learn

Now it's time to learn to classify text

Language of texts Español ▼

Learning to recognize text

3. Try

Introduces new terms and checks they are correctly classified

Expression

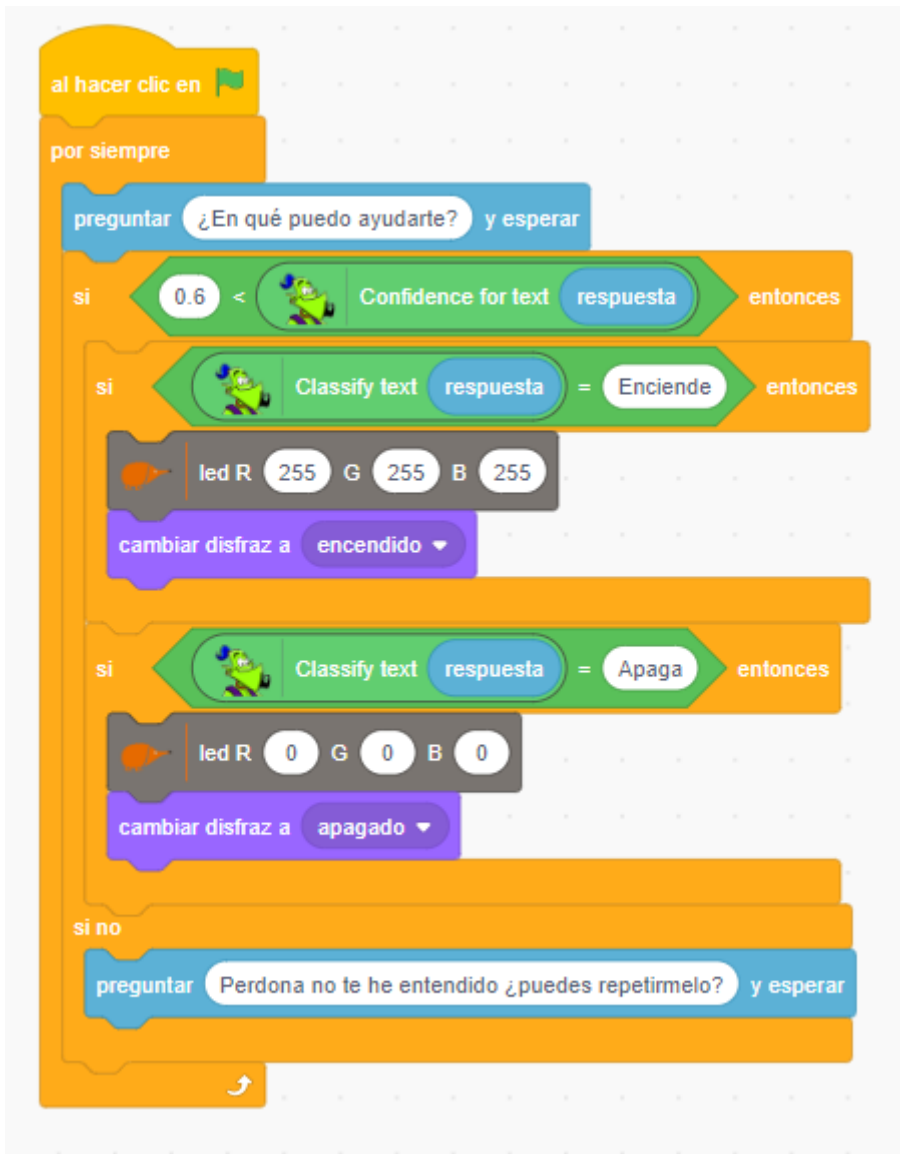
Mucha luz

Check

Believe belongs to the class Apaga, but I am not sure

- Apaga (50.75 %)
- Enciende (49.25 %)

**Ahora si !!** vamos a EchidnaScratch y creamos este programa usando las instrucciones de Learning ML que se basan en el modelo aprendido anteriormente



Aquí lo tienes todos los procesos en vídeo :

<https://www.youtube.com/embed/do0mDcAuE2w?t=134s>

CC-BY SA Fuente <https://echidna.es/a-programar/echidnaml/como-empezar-con-machine-learning-y-echidna/>

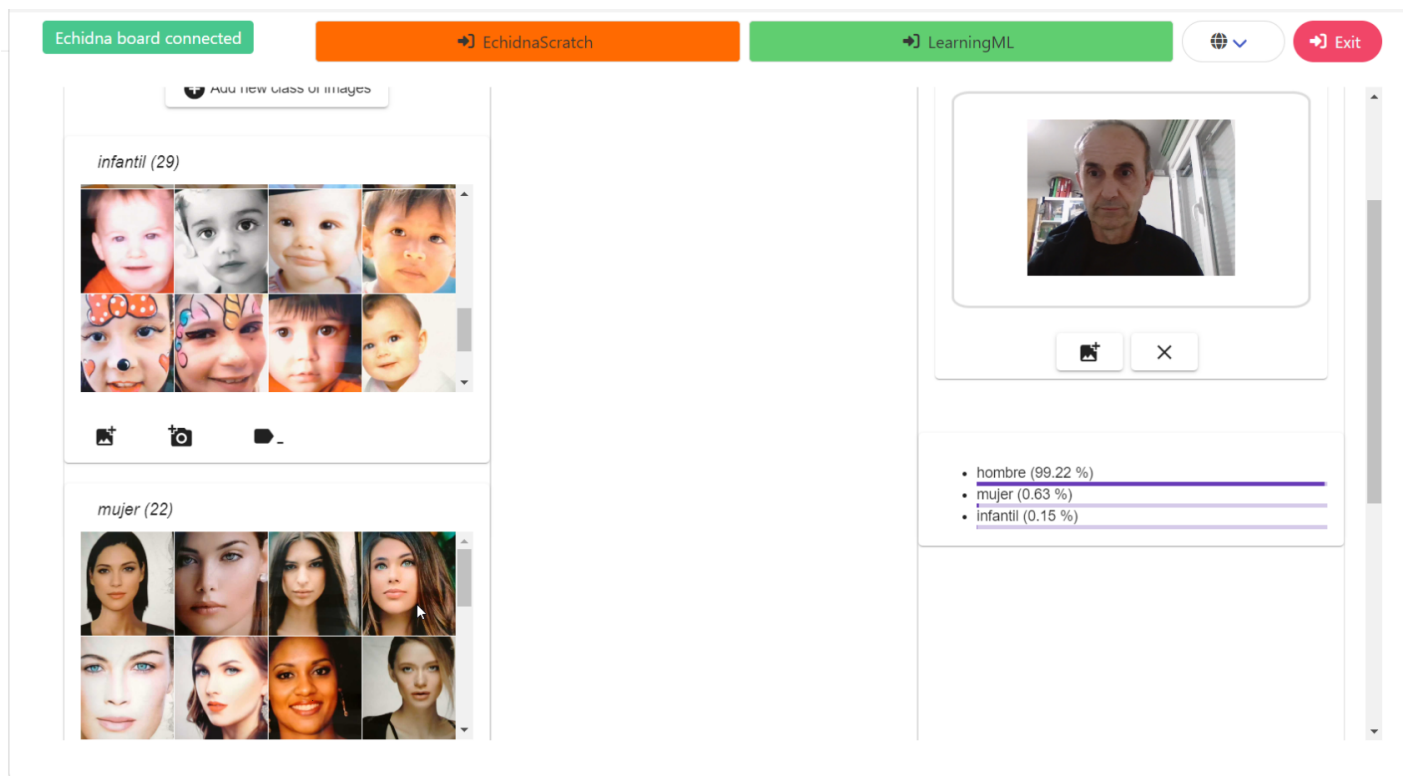
El programa en este repositorio: <https://github.com/JavierQuintana/Echidna>

El sprite lámpara lo he obtenido de <https://scratch.mit.edu/projects/198689957/editor/>

<https://www.youtube.com/embed/XDMYKfE46b8>

# MONTAJE 14 Reconocimiento de imágenes

El proceso es similar, en este caso he utilizado el móvil y he mostrado a la cámara web las búsquedas de caras para que aprenda la diferencia entre niños, hombres y mujeres

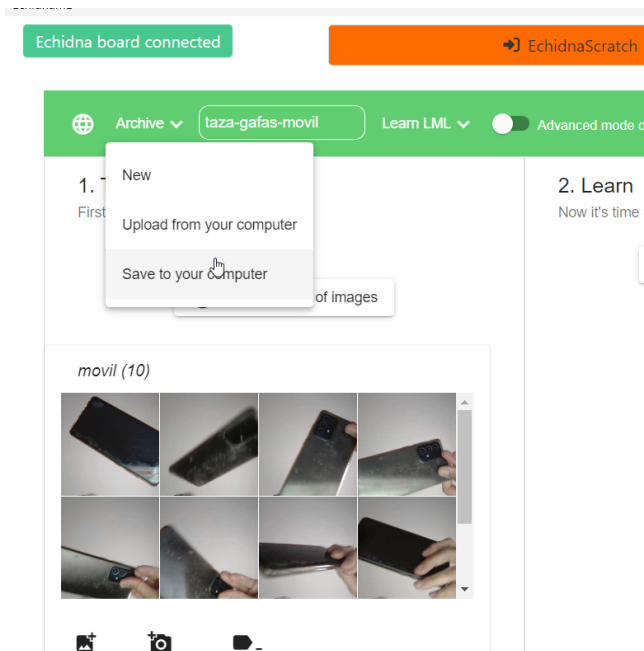


□□ como veis, para probar el modelo he utilizado mi cara y me reconoce un 99% de hombre□□□□□

## Grabar y abrir una máquina de IA

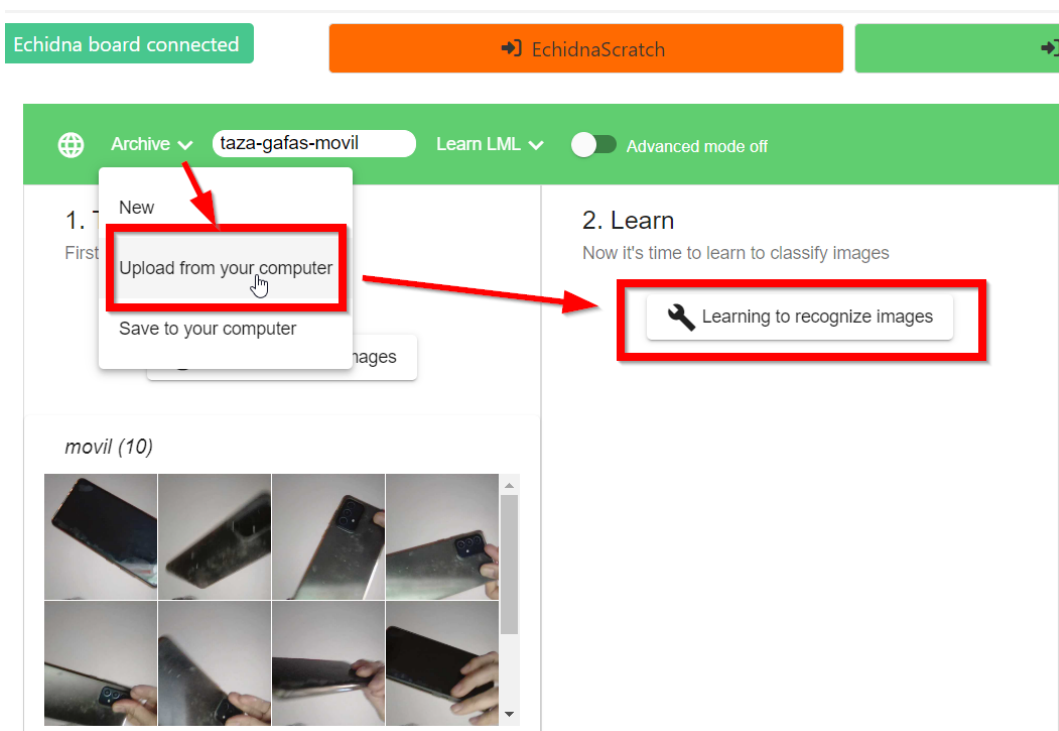
**OJO** hay que grabar el la máquina de inteligencia artificial que has creado !!! si cierras el programa se pierden todos los datos. **No advierte** que no lo has grabado

Voy a crear otra máquina IA con los objetos Gafas - Taza- móvil y lo grabo



Al grabar lo hace con un nombre de archivo .JSON lo voy a hacer con el nombre **taza-gafas-movil.json** que lo tienes en <https://github.com/JavierQuintana/Echidna>

Este archivo tiene **los datos de entrada NO LA MÁQUINA** luego si cerramos el programa y queremos abrirlo tenemos que volver a entrenarla:



## Programa en EchidnaScratch



## Resultado



<https://www.youtube.com/embed/n7UonllzHdQ>

Sin Echidna se puede hacer simplemente con un Arduino y tres leds, verde al D11 amarillo al D12 y el rojo al D13 ver <https://libros.catedu.es/books/echidna/page/tengo-arduino-puedo-hacer-los-programas-sin-echidna>

<https://www.youtube.com/embed/FILx9LxMQfQ>

## Recomendamos...

Ver todo el proceso en este vídeo

<https://www.youtube.com/embed/WRwzml9Dyns?t=261s>

Fuente <https://echidna.es/a-programar/echidnaml/como-empezar-con-machine-learning-y-echidna/>

## También recomendamos...

Usar **variables** para tener el control de lo que sucede de esta manera podemos controlar qué márgenes de confianza y cual es el resultado de la clasificación. Si el resultado no es el esperado:

- ajustar los valores de confianza en la imagen he puesto 0.9
- darle más fotos a la máquina de IA para que lo haga mejor.





**Reto :** Igual que los porteros automáticos, que se encienda la cámara si se aprieta el botón (por ejemplo SL)

**Solución:** En este caso es recomendable usar un pequeño retardo de 1segundo para que le de tiempo a capturar la imagen

Se puede mejorar el programa añadiendo un retardo antes de que se cierre la puerta, etc...



# MONTAJE 14BIS


## RECONOCIMIENTO DE MATRÍCULAS

Vamos a desarrollar este proyecto de Jorge Lobo @Lobo\_tic

[https://x.com/lobo\\_tic/status/1923996982516814064](https://x.com/lobo_tic/status/1923996982516814064)

“ —Antes, para salir del parking del supermercado, había que escanear el ticket de compra, pero ahora la barrera se abre sola como si supiera que ese coche puede salir.

—¿Queréis saber cómo lo sabe?

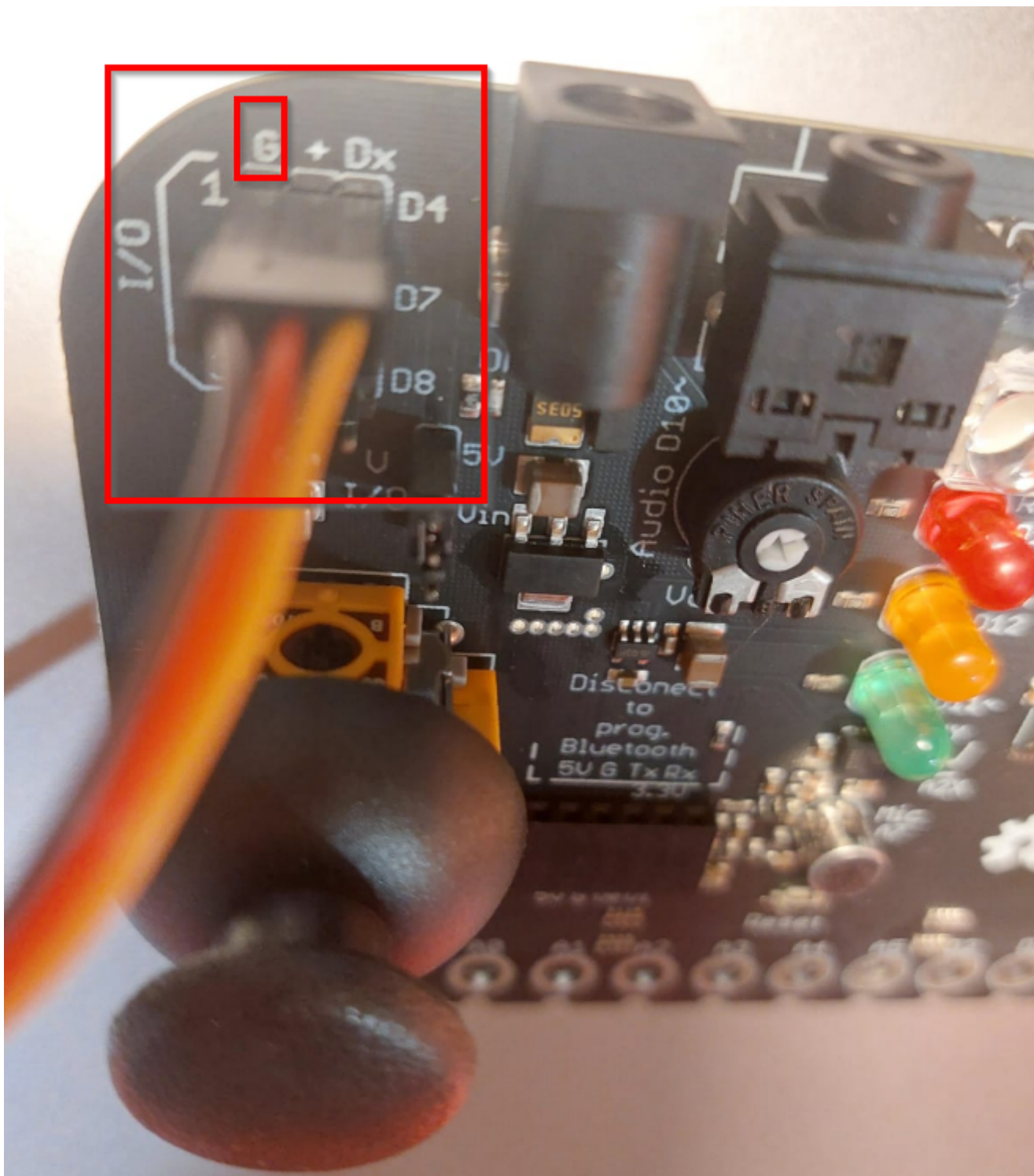
Así surge la idea de esta actividad con [@EchidnaSTEAM](#) y [@\\_learningml](#)   
[pic.twitter.com/3Ew22WcURj](https://pic.twitter.com/3Ew22WcURj)

— Jorge Lobo (@lobo\_tic) [May 18, 2025](#)

### PRIMER PASO : MONTAJE

Ponemos un servo en D4

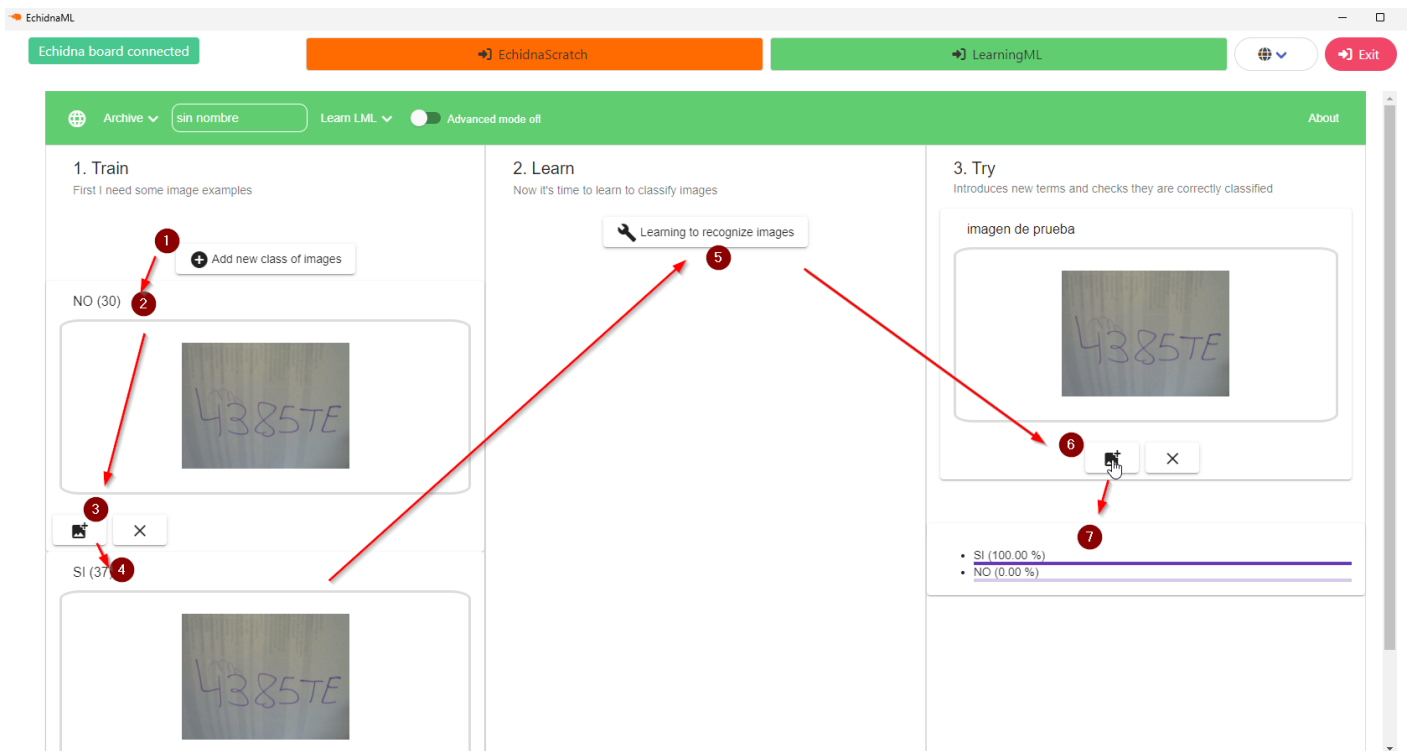
FÍJATE QUE EL **MARRÓN** A MASA GND SIMBOLO G TAL COMO LA FOTO Y EL AMARILLO EN DATO Dx



## SEGUNDO PASO : CREAR MÁQUINA IA DE RECONOCIMIENTO DE MATRICULAS

1. Entramos en crear máquina de reconocer imágenes
2. Creamos una clase que se llame **NO**
3. Añadimos imágenes desde nuestra webcam

1. Vamos enseñando matrículas que **no queremos que se abra el parking** de cada matrícula pulsamos una docena en diferentes posiciones (ligeros cambios)
2. También añadimos varias imágenes que existe cuando no hay ninguna matrícula, pues queremos que cuando no hay coche que no se abra el parking.
4. Añadimos la clase **SI**
  1. Repetimos el paso 3.1 con las matrículas que **queremos que se abra el parking**
5. Le damos a aprender
6. Probamos
7. Nos fijamos que funciona perfectamente



Detalle del paso 3.2

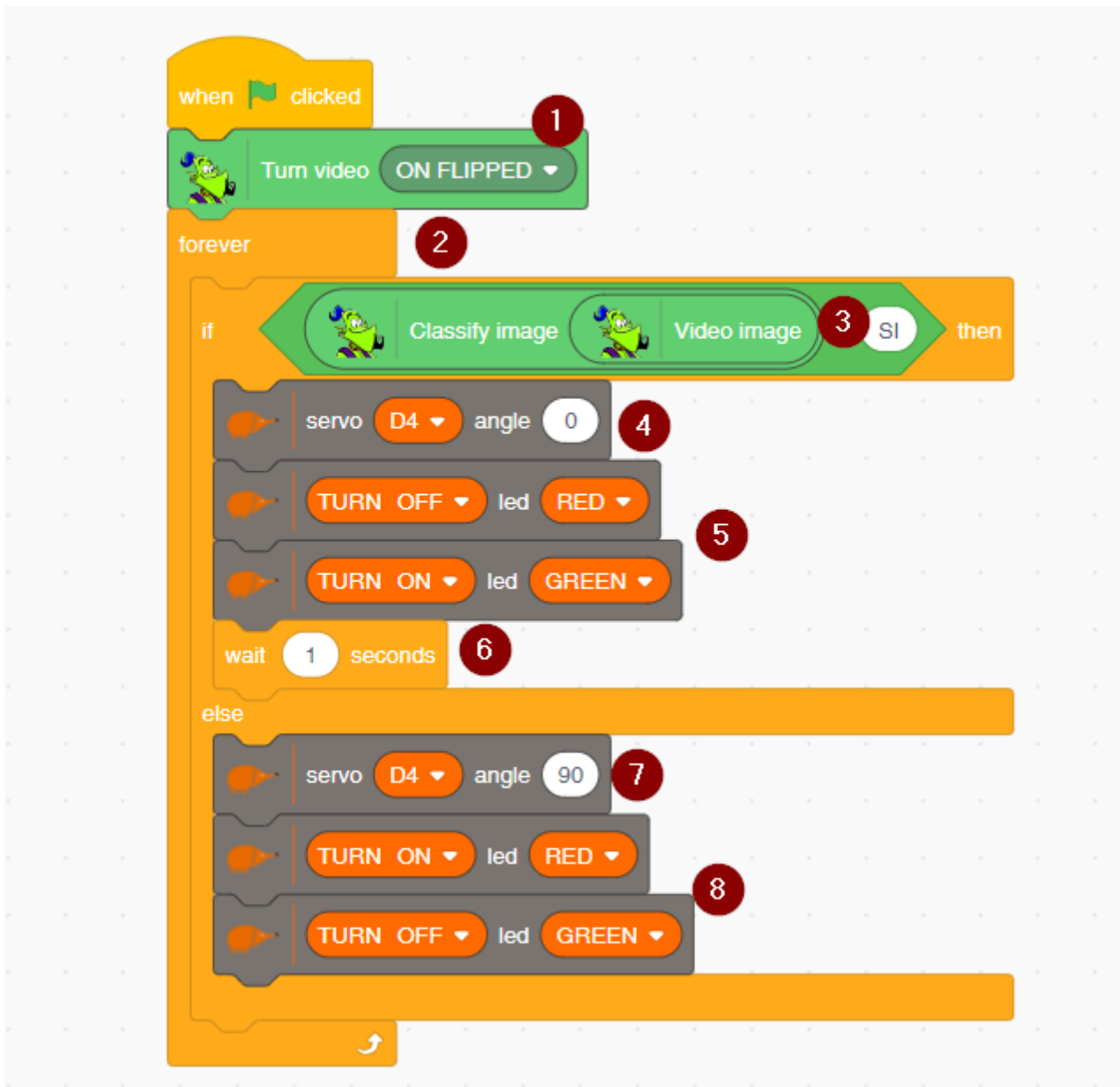
NO (30)



## TERCER PASO : PROGRAMAR

Ahora entramos en la ventana de programación usando el modelo:

1. En **learningml** arrastramos la instrucción de **encender cámara**
  1. curiosamente lo voltea, por lo tanto le decimos ON FLIPPED de lo contrario lee al revés las matrículas (curiosamente lo hace bien aunque a veces falla)
2. Ponemos un bucle infinito para que siempre lea los sensores y funcionen los actuadores (típico en programas de robótica)
3. Un condicional dentro del bucle si --- sino y en la condición que sea si LA MATRICULA ES DE LA CLASE **SI** pues que abra sino que la cierre
  1. Tiene dos instrucciones dentro de learningml uno que es que clasifique la imagen y otro que sea la imagen de la captura del vídeo
4. Si es SI pues que abra con el servo
5. También que encienda la luz verde y apague el rojo
6. Esta espera es para que el coche pase tranquilamente además que nos permite que no haga continuos aperturas y cierres de la barrera
7. SINO pues que cierre la barrera
8. Encienda rojo y apague verde



## CUARTO PASO : JUGAR

<https://www.youtube.com/embed/Bkvj0yK6t3A>