

Bloque 1: Introducción práctica

1.3. Paralelismo y eventos

José Ignacio Huertas Fernández

@jihuefer

Jesús Moreno León

@j_moreno1

<http://programamos.es>

info@programamos

Twitter: [@programamos](https://twitter.com/programamos)

<http://facebook.es/programamos>

©Programamos, octubre de 2015

Algunos derechos reservados. Este artículo se distribuye bajo la licencia "Reconocimiento-CompartirIgual 3.0 España" de Creative Commons, disponible en <http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>

Este documento (o uno muy similar) está disponible en (o enlazado desde) <http://programamos.es>

¿Qué vamos a ver?

- 1. Paralelismo**
- 2. Control de eventos**
- 3. Story telling en la educación.**
- 4. Otros eventos: los sensores**

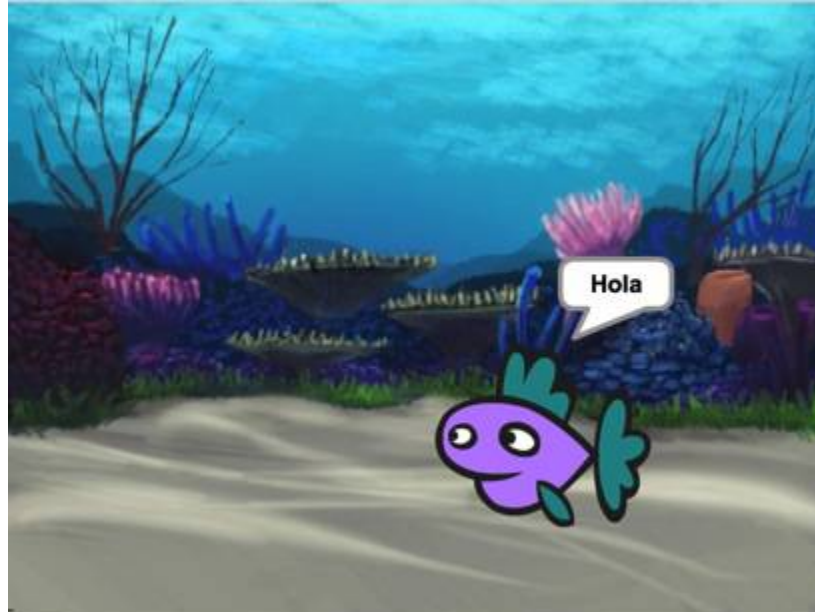
1. ¿Qué es el paralelismo?

- **Paralelismo:** podremos ejecutar distintas líneas de código en paralelo, ya sea en el mismo o bien en otro objeto.



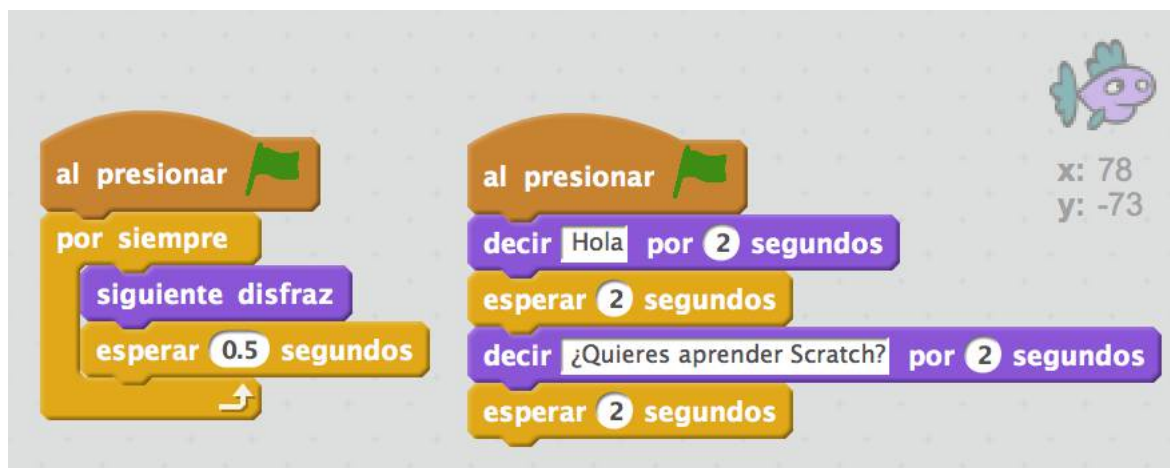
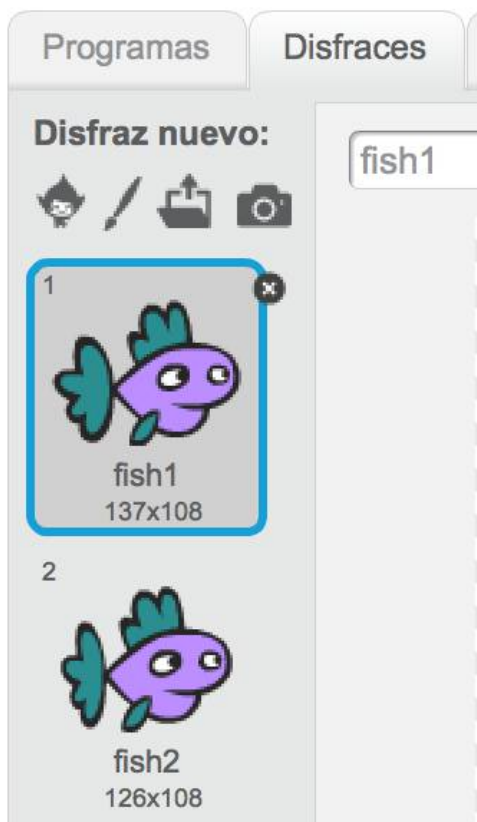
2. Practicamos: mover y hablar

- Para darle más realismo a nuestro personaje vamos a darle algo de movimiento a la vez que va hablando.
- Haz que el pez mueva la cola mientras dice algo.



2. Practicamos: mover y hablar

- Una posible solución:



3. Sincronización de procesos

- Para sincronizar distintas líneas de ejecución, ya sea en el mismo objeto o bien en distintos, podemos hacer uso de la instrucción “esperar”.
- Más adelante veremos otra solución con envío de mensajes.



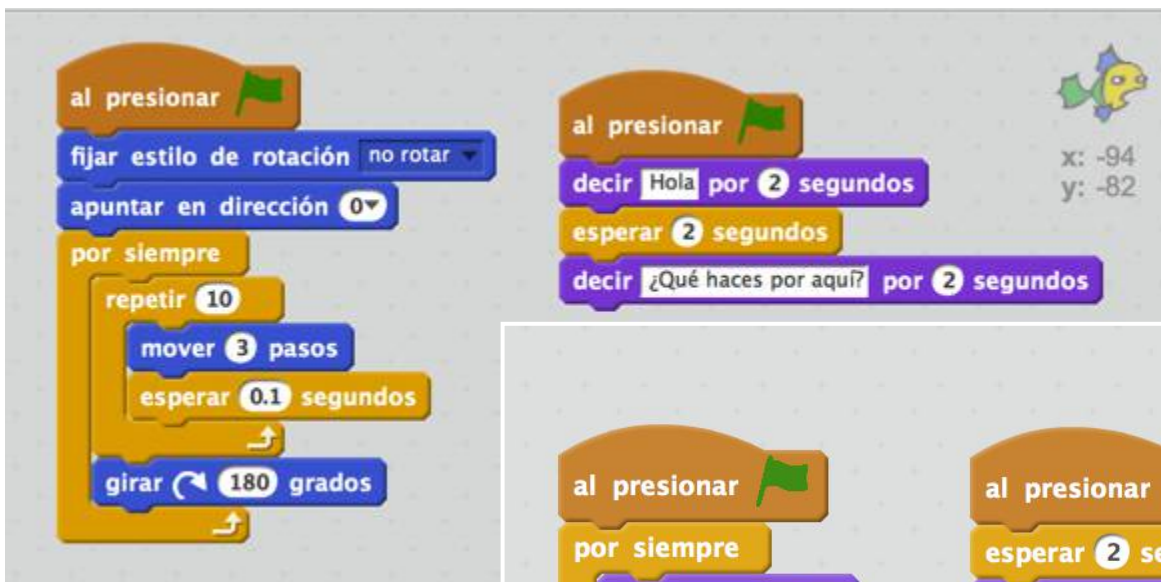
4. Practicamos: diálogo

- Realiza un diálogo entre dos personajes.
- Haz que los peces se muevan mientras hablan.




4. Practicamos: diálogo

- Una posible solución.



Scratch script for a yellow fish character (coordinates: x: -94, y: -82):

- al presionar 
- fijar estilo de rotación no rotar
- apuntar en dirección 0
- por siempre
 - repetir 10
 - mover 3 pasos
 - esperar 0.1 segundos
 - girar 180 grados



Scratch script for a purple fish character (coordinates: x: 78, y: -73):

- al presionar 
- por siempre
 - siguiente disfraz
 - esperar 0.5 segundos

Parallel script:

- al presionar 
- esperar 2 segundos
- decir Hola por 2 segundos
- esperar 2 segundos
- decir Pues ¡aprender Scratch! por 2 segundos
- esperar 2 segundos

1. ¿Qué son los eventos?. Tipos

- **Eventos:** dictan cuándo se empieza a ejecutar código. Podemos distinguir dos tipos:

- Propios de Scratch
- Personalizados

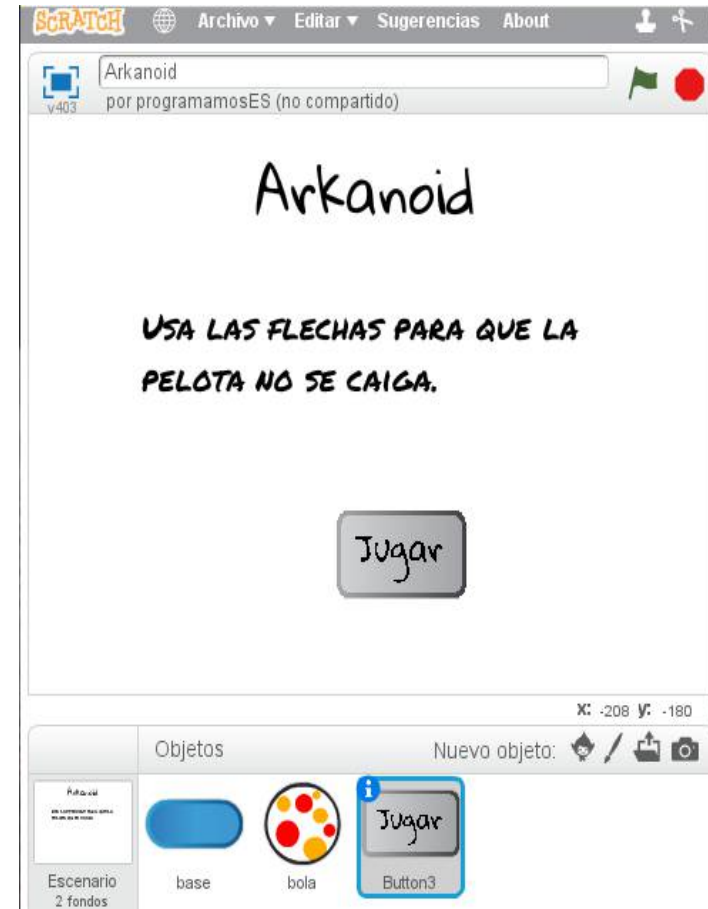


2. Envío de mensajes

- Son instrucciones incluidas en el bloque eventos que permiten crear un evento y capturarlo, ya sea en el mismo objeto o bien en otros (incluso en el escenario).
- Permiten sincronizar procesos.

3. Practicamos

- Vamos a realizar una pantalla de inicio para el videojuego Arkanoid.
 - Al pulsar sobre la bandera verde se debe mostrar un fondo con el título y descripción del funcionamiento del juego y un botón para comenzar a jugar.
 - El resto de objetos inicialmente estarán ocultos y se mostrarán después de haberse pulsado el botón jugar. También se cambiará el fondo y se ocultará el botón.



3. Practicamos

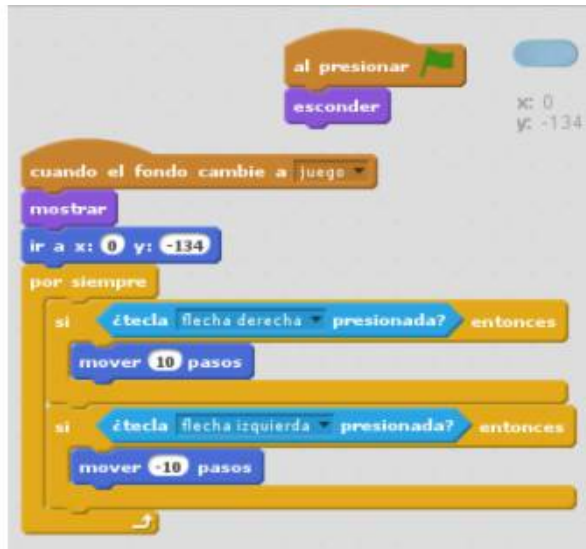
- Una posible solución.

Escenario



```
al recibir aJugar
  cambiar fondo a juego

al presionar
  cambiar fondo a principal
```



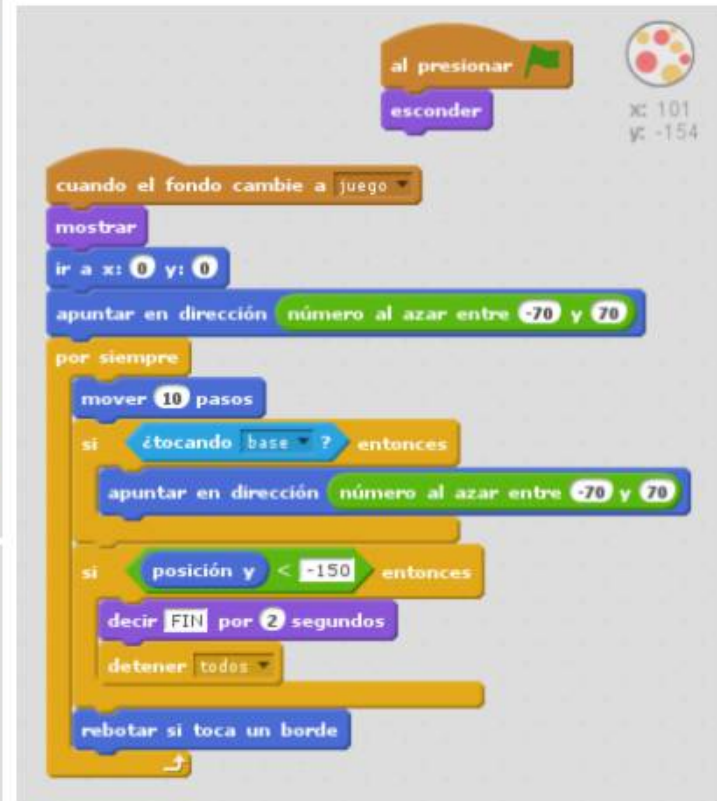
```
al presionar
  esconder

cuando el fondo cambie a juego
  mostrar
  ir a x: 0 y: -134
  por siempre
    si ¿tecla flecha derecha presionada? entonces
      mover 10 pasos
    si ¿tecla flecha izquierda presionada? entonces
      mover -10 pasos
```



```
al presionar
  ir a x: 19 y: -96
  mostrar

al clicar este objeto
  cambiar disfraz a button3-b
  esperar 0.2 segundos
  cambiar disfraz a button3-a
  esperar 1 segundos
  enviar aJugar
  esconder
```



```
al presionar
  esconder

cuando el fondo cambie a juego
  mostrar
  ir a x: 0 y: 0
  apuntar en dirección número al azar entre -70 y 70
  por siempre
    mover 10 pasos
    si ¿tocando base? entonces
      apuntar en dirección número al azar entre -70 y 70
    si posición y < -150 entonces
      decir FIN por 2 segundos
      detener todos
  rebotar si toca un borde
```


1. ¿Qué son?

- Son **proyectos narrativos**, en los que se cuenta una historia.
- Son muy utilizados en Scratch.
- Algunos ejemplos:
 - En clase de **idiomas**: Contar una historia escribiendo (o diciendo) los diálogos en el idioma correspondiente en los que tengan que incluir el vocabulario del tema que se esté trabajando.
 - En clase de **historia**: Contar una historia, incluyendo personajes históricos y haciendo un trabajo de documentación de la época que se esté trabajando.
 - En **literatura**: dando vida a un libro o poema que se haya leído en clase.
 - En **matemáticas**: explicando las fracciones, por ejemplo.
 - En **física**: explicando alguna ley como la gravedad, pero de forma divertida o interactiva.



2. ¿Cuál es su aplicación en la educación?

- Permiten que, a través de la programación, se trabajen distintas áreas: lingüística, historia, conocimiento del medio, etc.
- Cualquier docente de cualquier especialidad puede usar Scratch para crear este tipo de proyectos en su aula como actividad.
- ¿Ventajas?
 - Permite al alumnado expresarse
 - Creatividad
 - Motivación y satisfacción
 - Aprenden creando algo suyo
 - Usan la tecnología para aprender
 - Es divertido

3. Practicamos: crea tu propia historia

- Puedes aplicarla a cualquier área y temática.



<http://scratch.mit.edu/projects/22810116>

1. Los sensores en Scratch

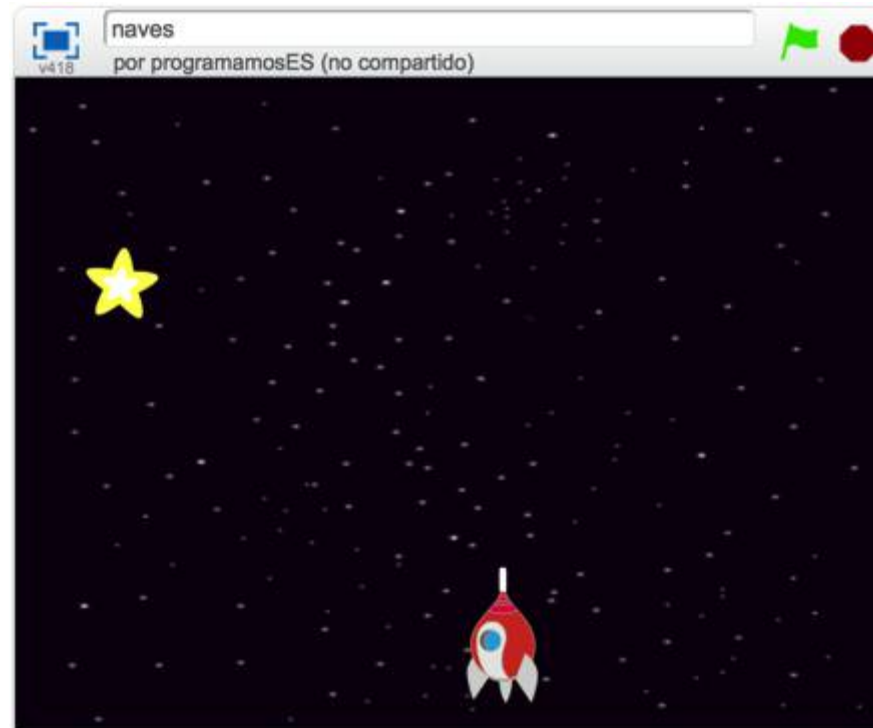
- Nos van a permitir interactuar con el usuario, capturando eventos de:
 - Pulsaciones de teclado
 - Movimiento de ratón
 - Intensidad del sonido
 - Movimiento en el vídeo
 - Dirección en el vídeo
- También permiten:
 - Preguntar y recoger las respuestas.
 - Interacción entre distintos objetos: distancia, tocando un color...



Otros eventos: los sensores

2.1 Ejemplo de uso: nave

- Con el teclado o el ratón.

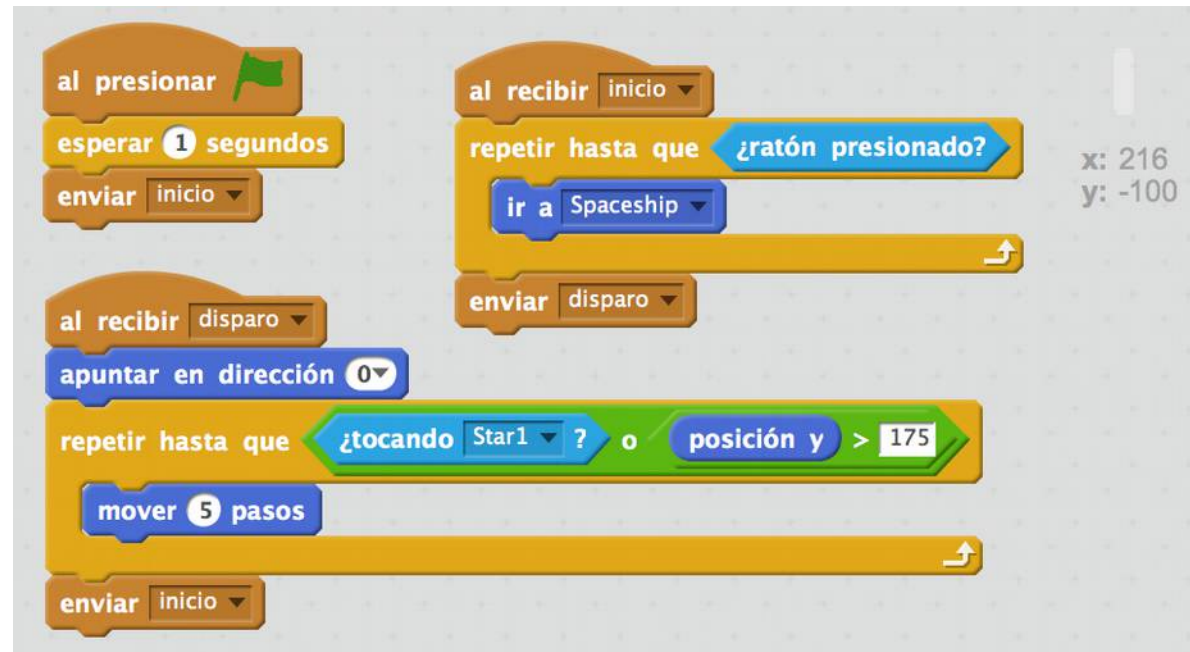
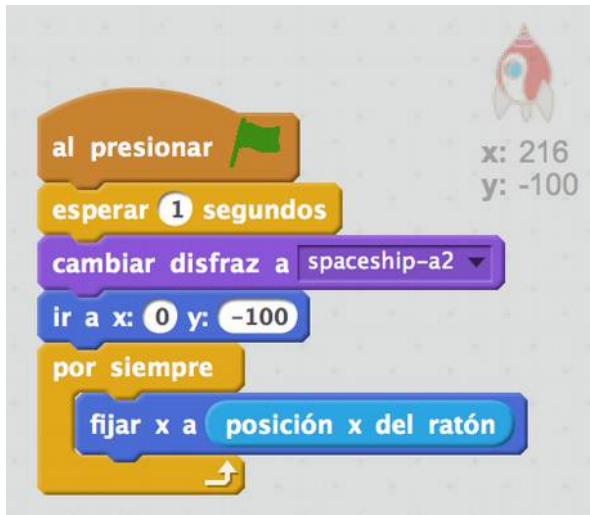


<http://scratch.mit.edu/projects/22866671/>

Otros eventos: los sensores

2.1 Ejemplo de uso: nave

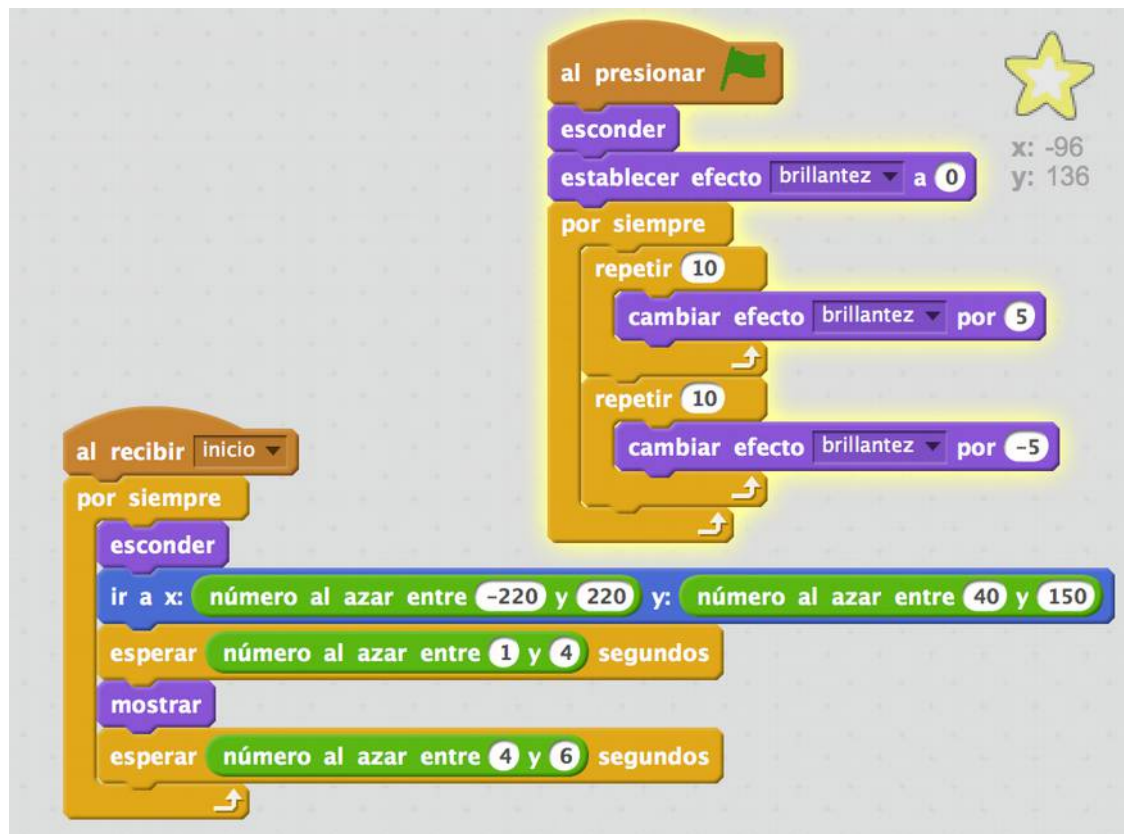
- Una posible solución



Otros eventos: los sensores

2.1 Ejemplo de uso: nave

- Una posible solución.



Otros eventos: los sensores

2.2 Ejemplo de uso: el dragón y el gato

- Con distancias, tocando objeto y color.



<http://scratch.mit.edu/projects/22865646/>

Otros eventos: los sensores

2.2 Ejemplo de uso: el dragón y el gato

- Una posible solución.



The image shows a Scratch script for a dragon character. The script is as follows:

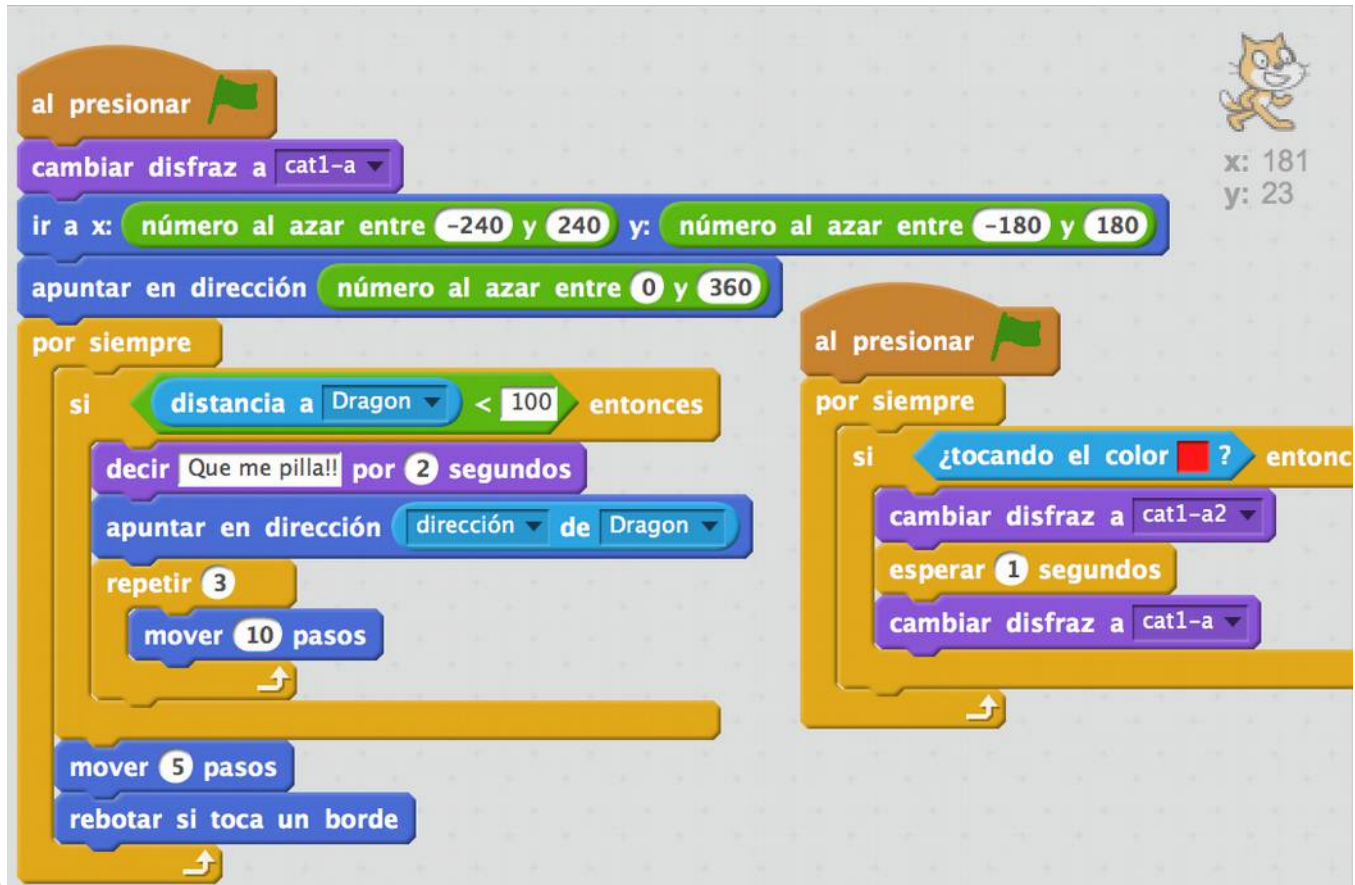
```
al presionar [bandera verde]
  ir a x: número al azar entre -240 y 240 y: número al azar entre -180 y 180
  esperar 1 segundos
  por siempre
    apuntar hacia gato
    mover 2 pasos
    si [¿tocando gato?] entonces
      decir Te cogí por 2 segundos
```

Coordinates: x: 95, y: 15

The script is designed to move the dragon to a random position and then check if it is touching the cat. If it is, it will say "Te cogí" for 2 seconds.

2.2 Ejemplo de uso: el dragón y el gato

- Una posible solución.



The image shows a Scratch script for a cat character. The script is divided into two main sections, each starting with an 'al presionar' (when clicked) event block.

Left Section:

- al presionar** (when clicked)
- cambiar disfraz a** (change costume to) `cat1-a`
- ir a x:** `número al azar entre -240 y 240` **y:** `número al azar entre -180 y 180`
- apuntar en dirección** (point in direction) `número al azar entre 0 y 360`
- por siempre** (forever loop):
 - si** (if) `distancia a Dragon < 100` **entonces** (then):
 - decir** (say) `Que me pilla!!` **por 2 segundos** (for 2 seconds)
 - apuntar en dirección** (point in direction) `dirección de Dragon`
 - repetir 3** (repeat 3 times):
 - mover 10 pasos** (move 10 steps)
 - mover 5 pasos** (move 5 steps)
 - rebotar si toca un borde** (bounce if touches edge)

Right Section:

- al presionar** (when clicked)
- por siempre** (forever loop):
 - si** (if) `¿tocando el color [red]?` **entonces** (then):
 - cambiar disfraz a** (change costume to) `cat1-a2`
 - esperar 1 segundos** (wait 1 seconds)
 - cambiar disfraz a** (change costume to) `cat1-a`

Coordinates: x: 181, y: 23