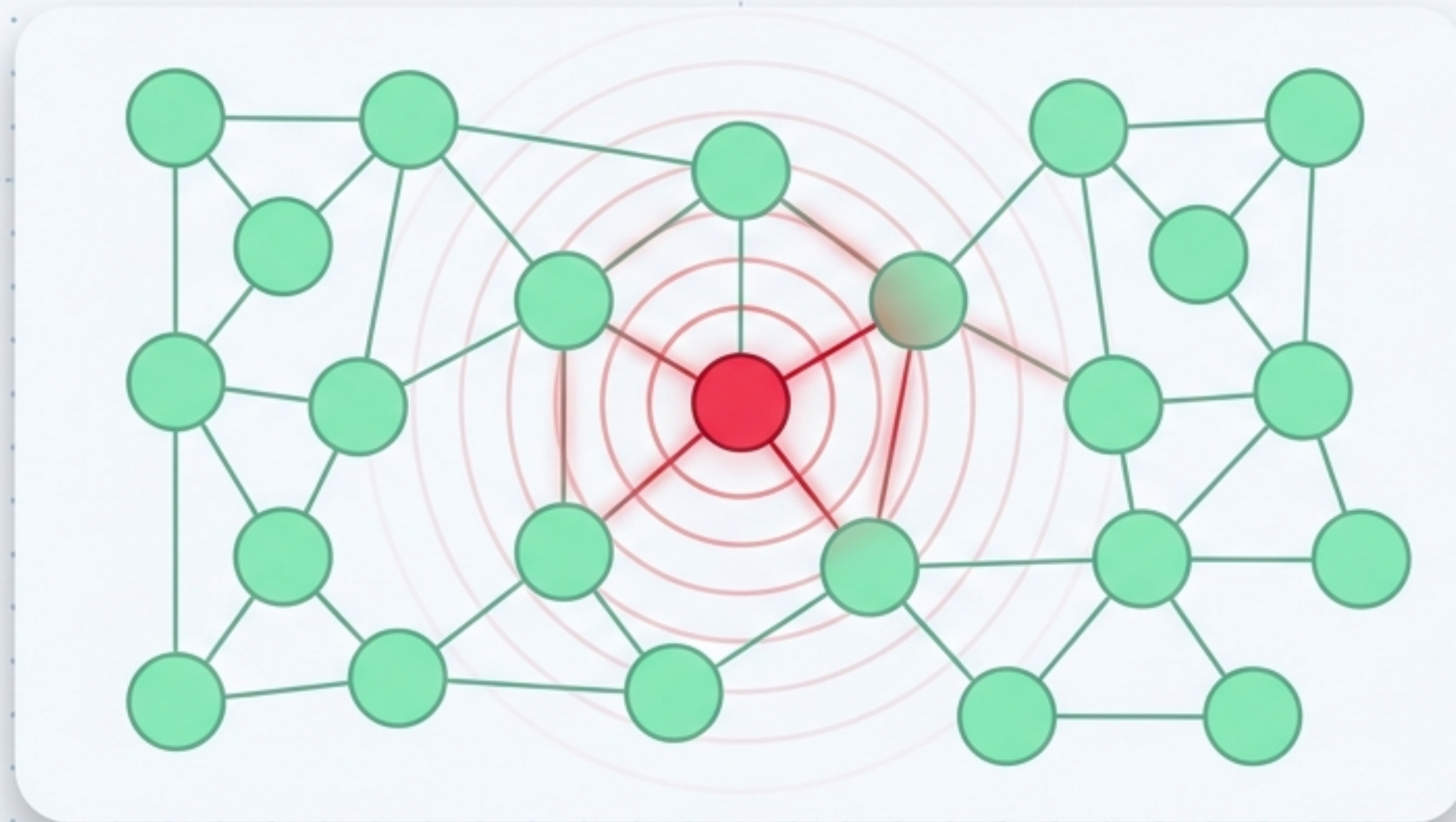


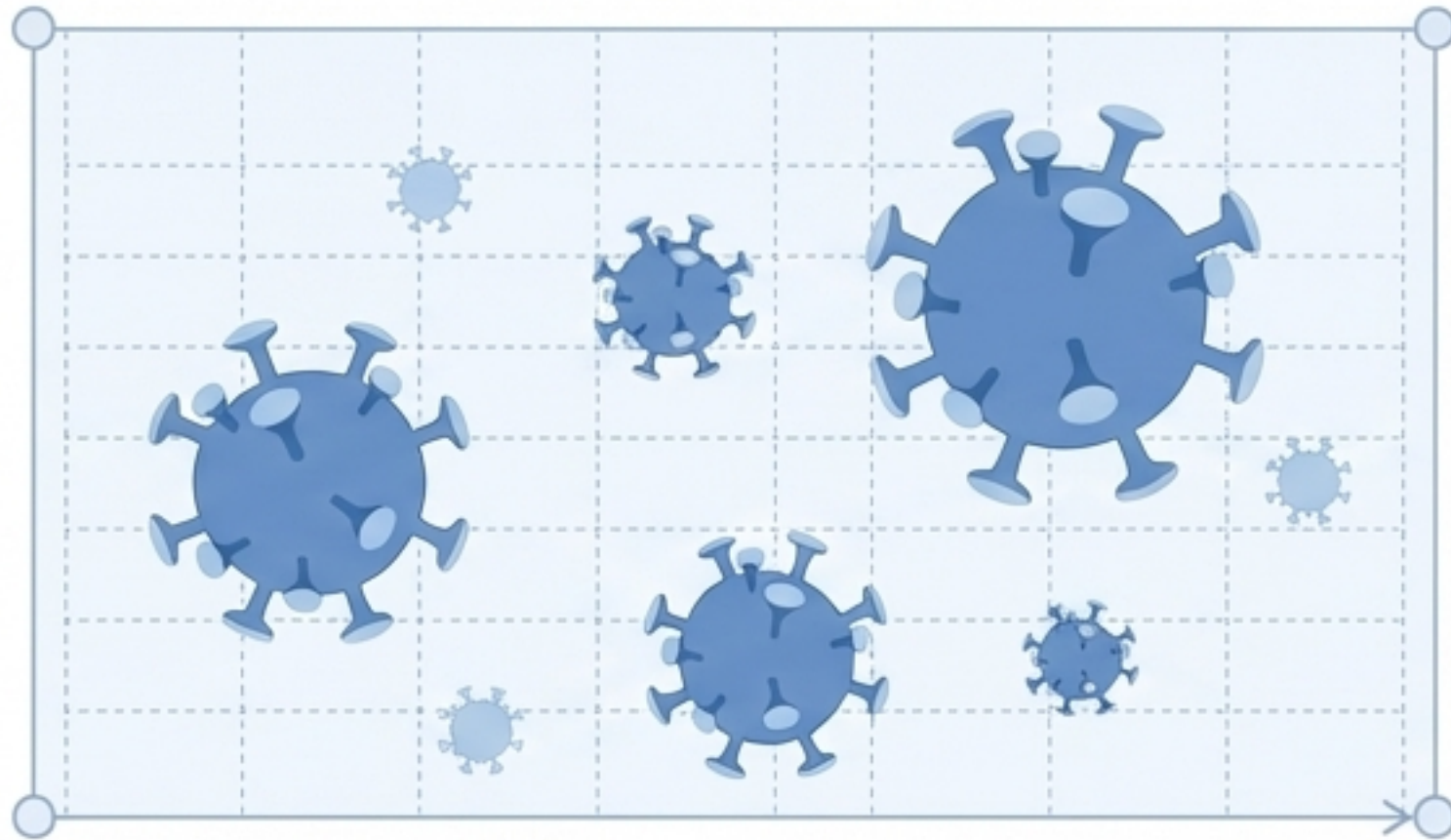
Simulando lo Invisible: Introducción a la Programación con Scratch

Guía de Laboratorio Computacional para Modelar Sistemas Complejos



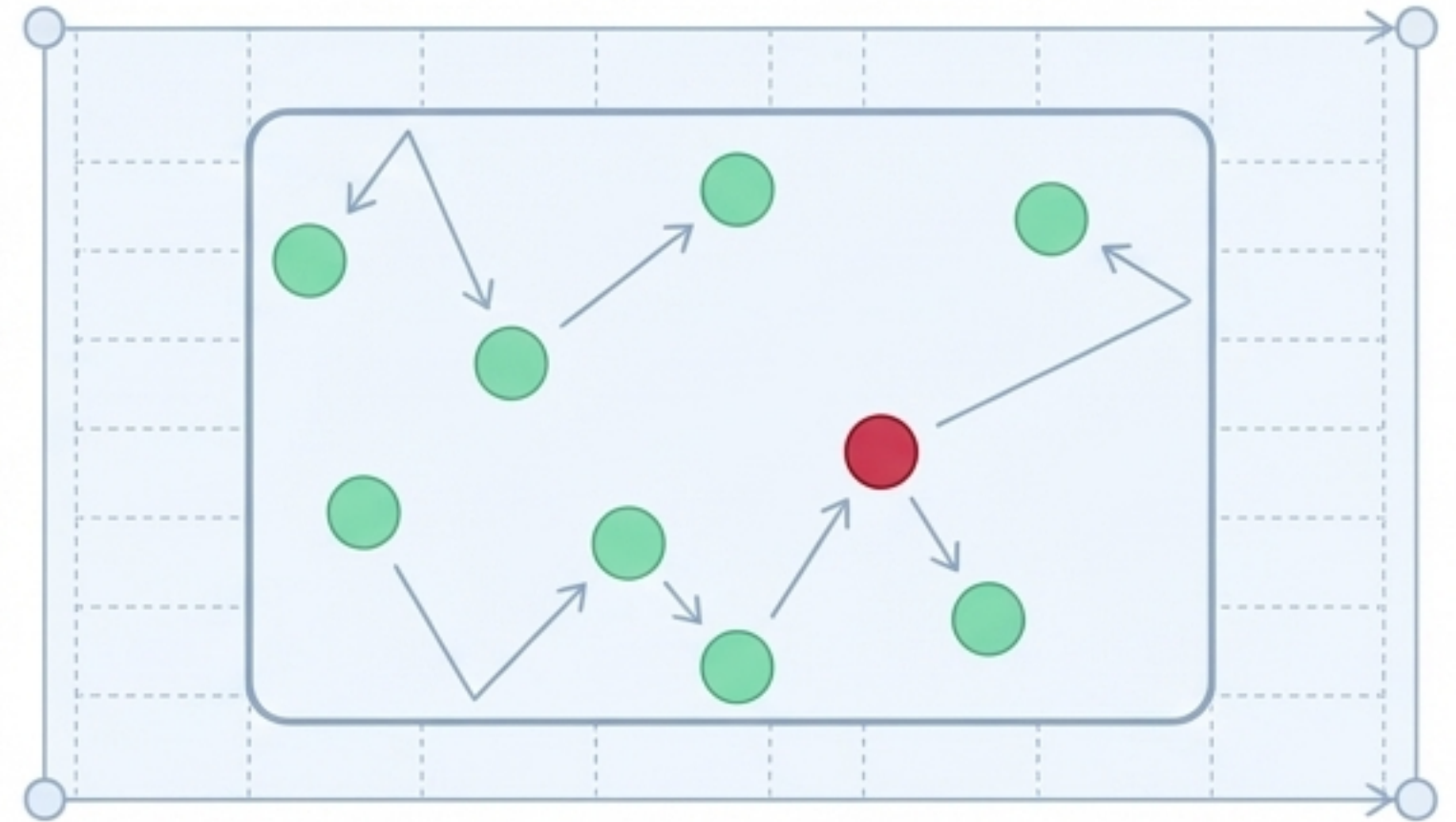
¿Cómo se propaga un virus en una comunidad?

El Desafío Real



En el mundo real, los virus son invisibles y su propagación es rápida. Es difícil prever cómo interactuarán miles de personas en movimiento constante.

La Solución Digital



Las simulaciones computacionales nos permiten crear 'laboratorios virtuales'. Modelamos fenómenos complejos de forma visual e interactiva para tomar decisiones informadas.

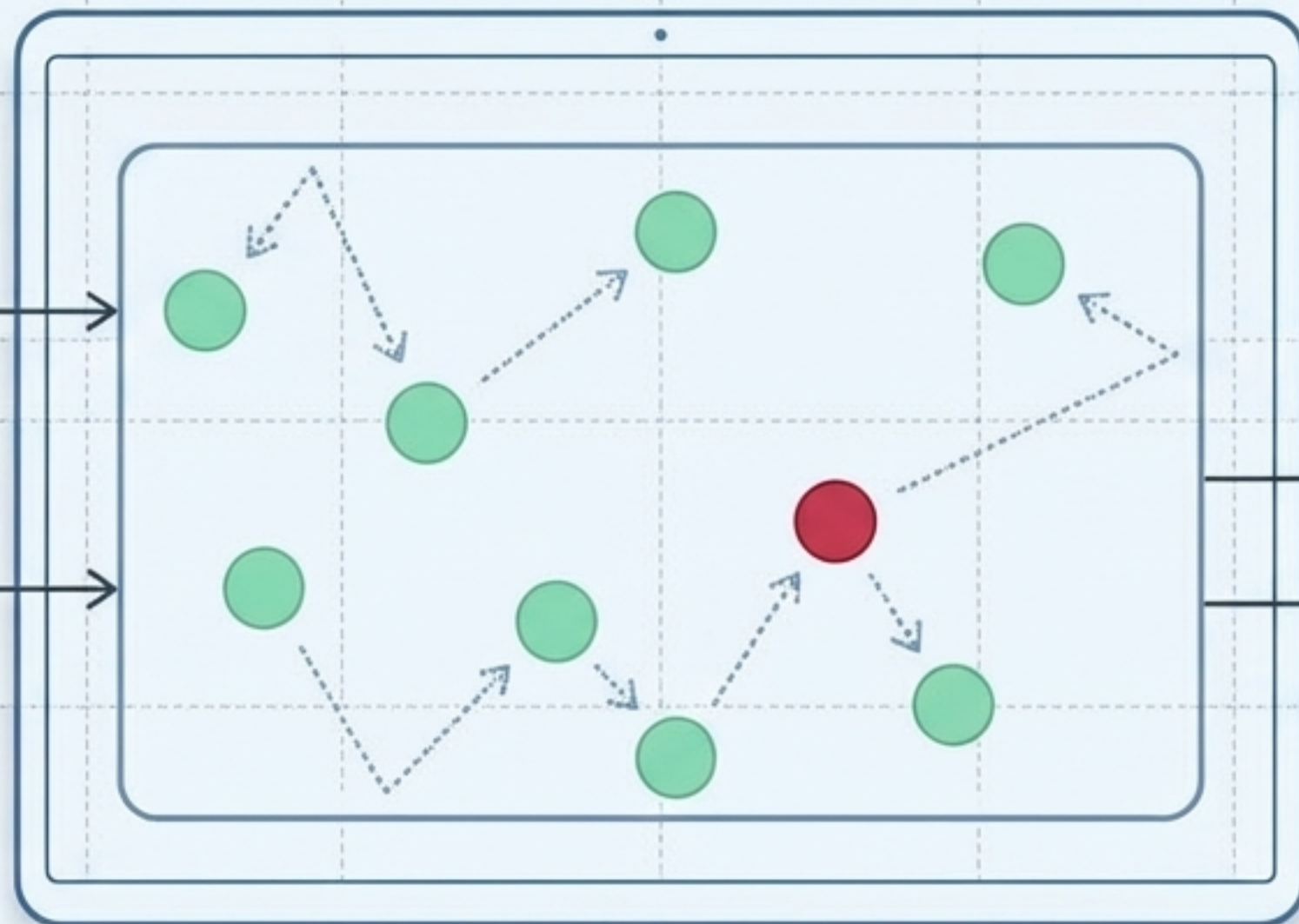
El Reto Principal: Diseñar el Modelo de Contagio

Agentes

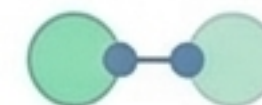
Personas sanas (**Círculos Verdes**) y 1 Paciente Cero (**Círculo Rojo**).

Física

Movimiento aleatorio constante. Si tocan el borde, deben rebotar.



Interacción

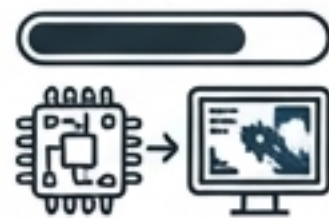
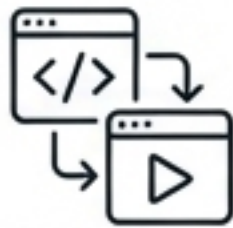


El contagio ocurre por contacto físico en el escenario.

Transformación

Cambio de estado (**Verde** cambia a **Rojo** al interactuar).

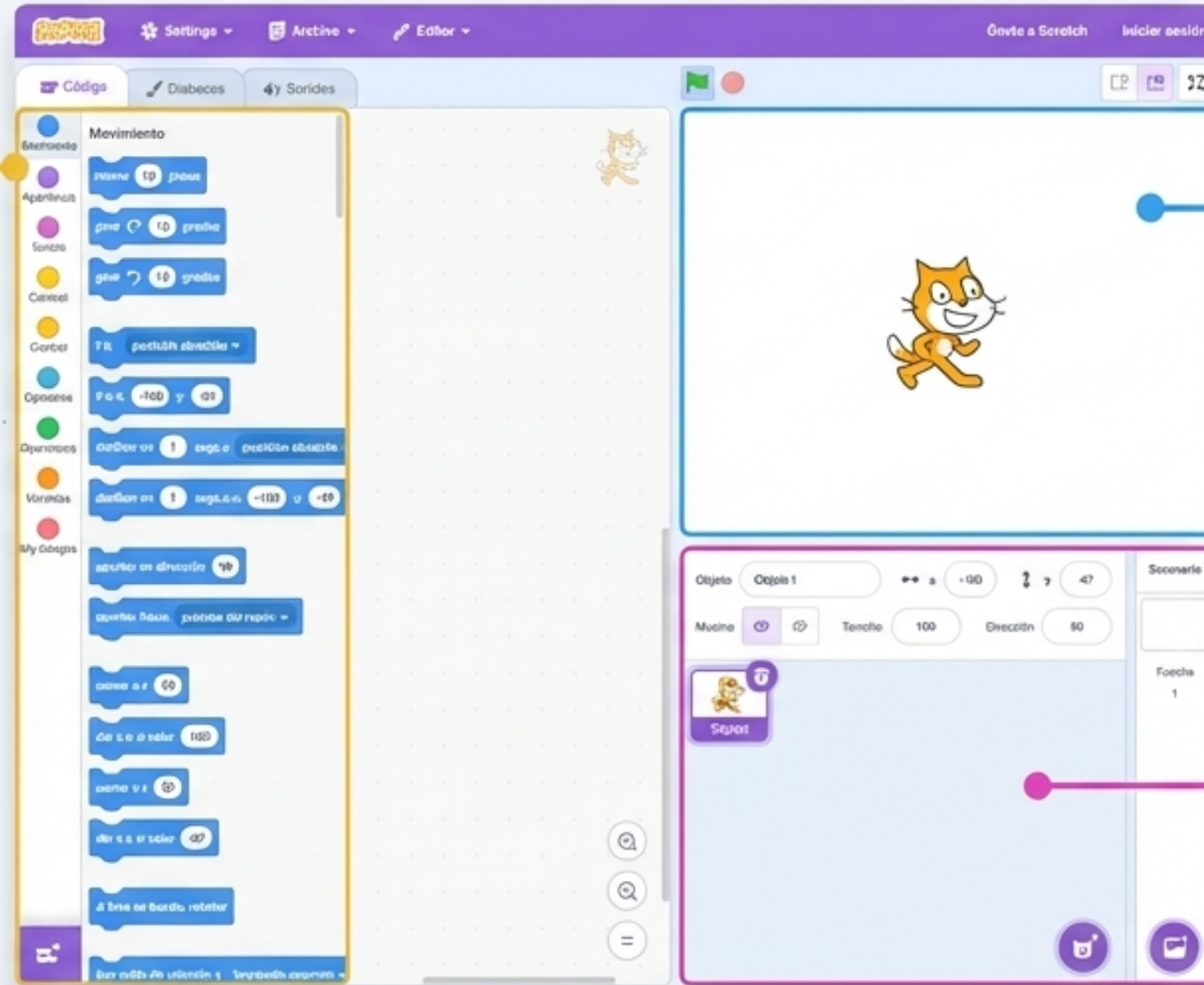
Scratch: Nuestro Motor de Simulación Lógica



- **Origen:** Desarrollado por el MIT Media Lab como una herramienta de experimentación algorítmica.
- **Función:** Transforma la lógica matemática en acciones visuales inmediatas mediante bloques gráficos encajables.
- **Evolución:** Si en misiones anteriores usaste lenguajes de bloques para hardware, Scratch nos permitirá orquestar escenarios digitales completos y dinámicos.

Anatomía del Entorno de Desarrollo

Zona 1: Área de Código.
El inventario de comandos. Aquí extraemos las instrucciones mecánicas, sensoriales y lógicas.



Zona 2: El Escenario.
El espacio de simulación donde los algoritmos y los agentes cobran vida.

Zona 3: Paleta de Objetos.
Los 'actores' de nuestro modelo. Aquí añadimos o modificamos las propiedades físicas.

Pensamiento Computacional: La Secuencia de Ejecución

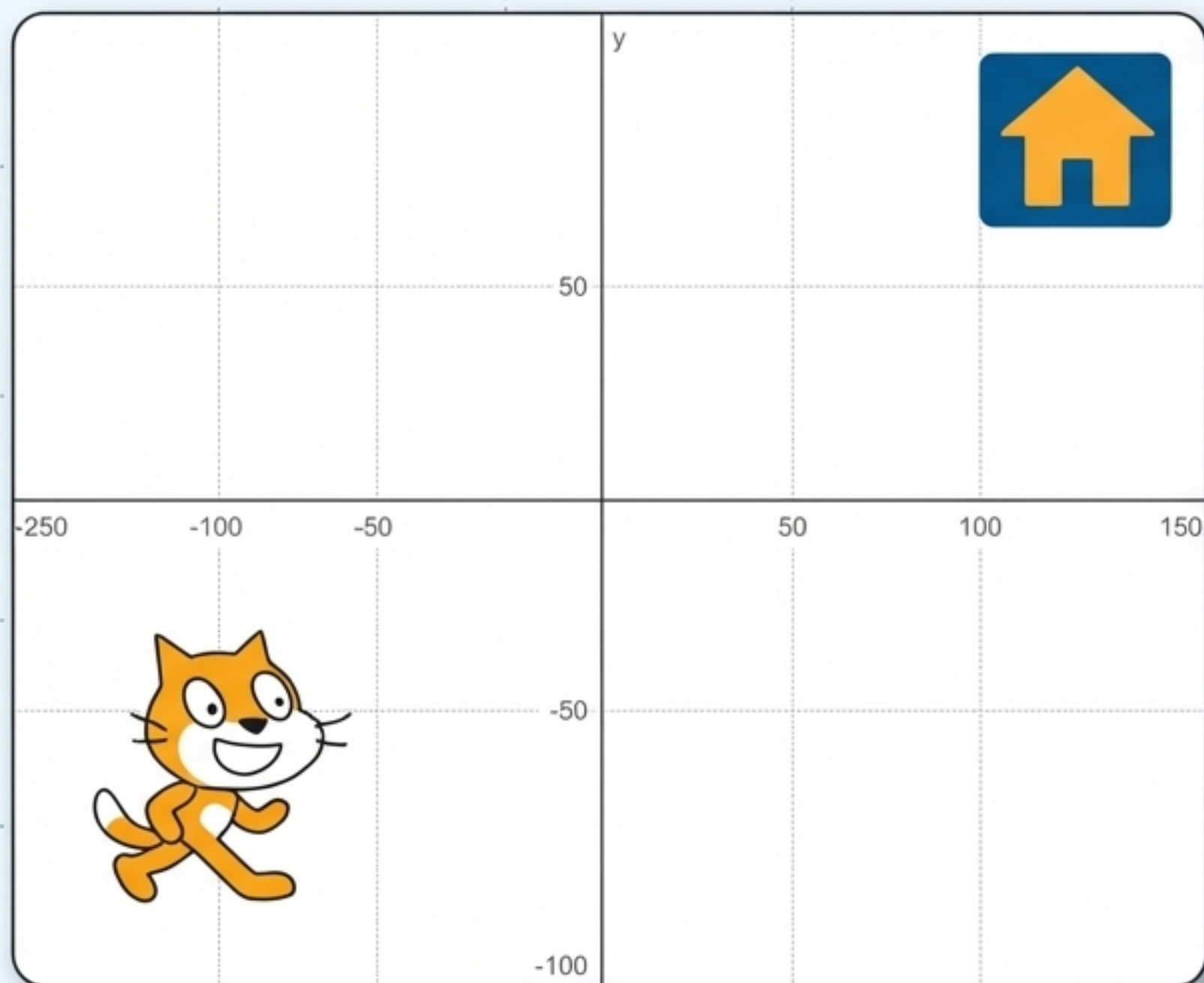


Instrucciones Aisladas
(Inactivas)

Algoritmo Secuencial
(Activo)

El orden es crítico. El computador ejecutará las acciones estrictamente de arriba hacia abajo. Un cambio en el orden cambia completamente el resultado de la simulación.

Misión de Entrenamiento: Prueba de Movimiento



Objetivo del Experimento:

Programar un agente piloto (Gato) para que se desplace autónomamente por el escenario y llegue a una coordenada objetivo (Casa) al recibir un estímulo inicial (Clic).

Por qué hacemos esto:

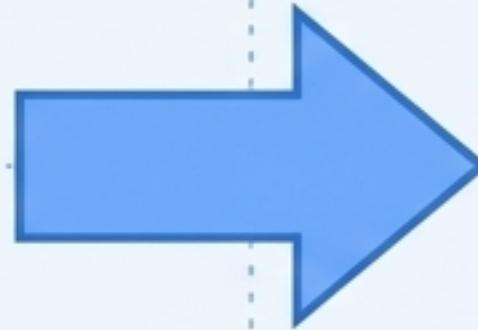
Dominar el evento de inicio y la mecánica de desplazamiento son los primeros pasos técnicos obligatorios para poder construir el movimiento aleatorio incontrolable del virus más adelante.

Codificando el Desplazamiento

al hacer clic en



CAUSA



EFEECTO

deslizar en 1 segs a Home Button ▾

El Gatillo (Evento):
Esto le dice al computador cuándo iniciar la secuencia.

La Acción (Movimiento):
Calcula automáticamente la ruta matemática desde la posición actual hasta el objetivo y ejecuta el desplazamiento en 1 segundo.

Al unir físicamente ambas piezas, hemos creado un programa funcional.
Prueba ejecutando la bandera verde.

Añadiendo Retroalimentación Sensorial

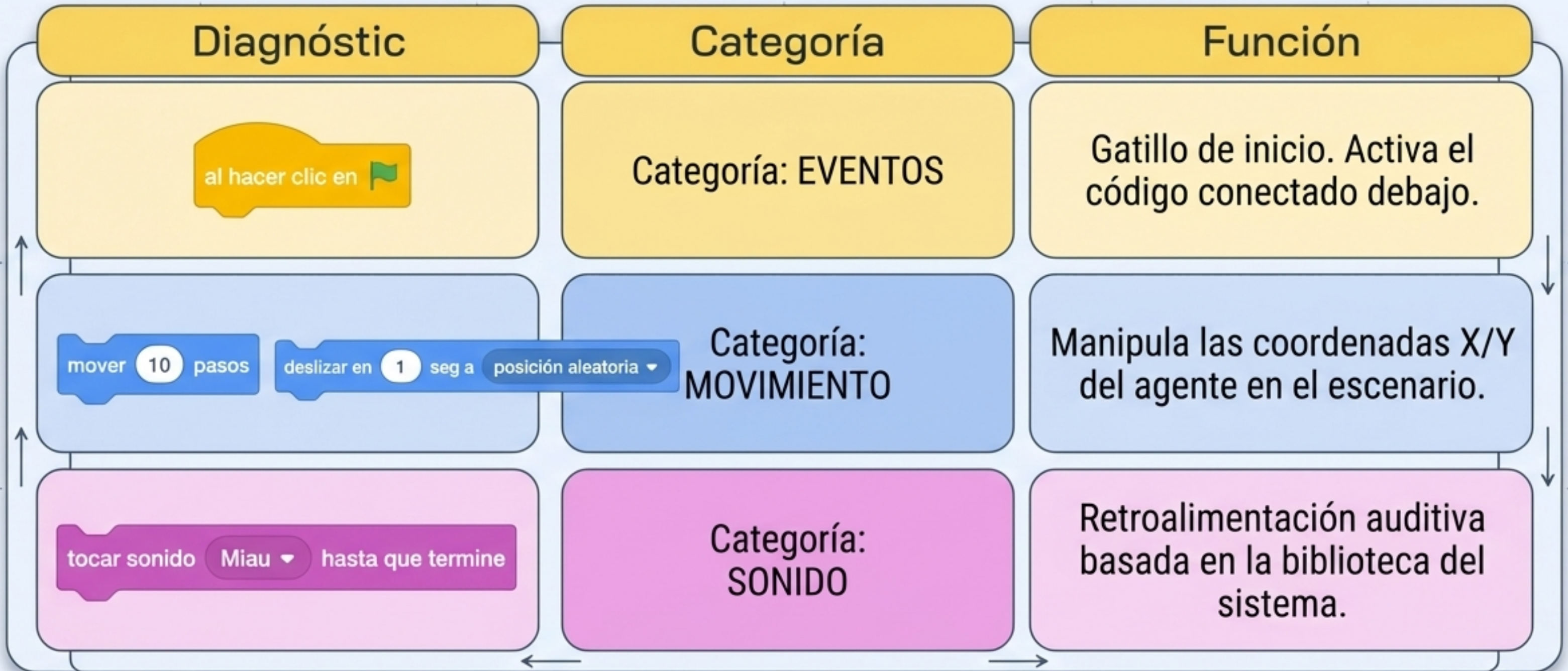


La Importancia del Finalizador

El sonido Miau se ejecutará únicamente después de que el deslizamiento de 1 segundo se haya completado al 100%.

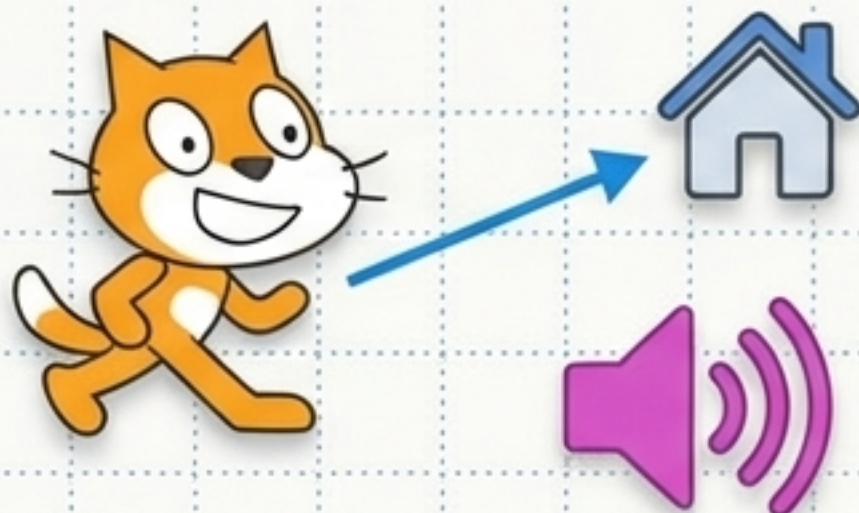
¿Por qué? La computadora procesa el código línea por línea. El comando "hasta que termine" fuerza al sistema a pausar el avance a la siguiente línea de código hasta que la acción actual concluya.

Matriz de Bloques de Programación



Del Entrenamiento al Reto Final

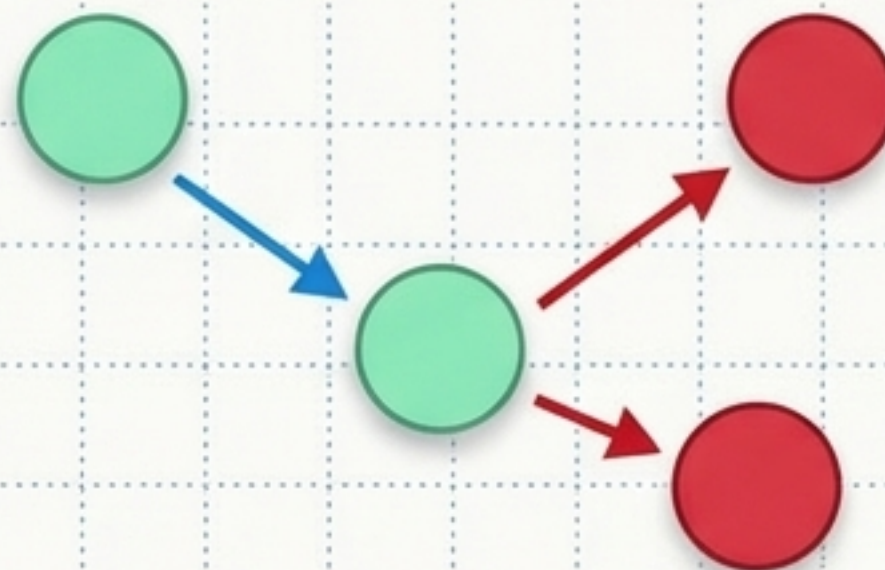
Lo que ya dominas



Agente se desliza a un objetivo
→ Detecta contacto.

=

Lo que construirás pronto



Agente se desliza aleatoriamente
→ Detecta contacto viral.

El **bloque de deslizamiento** es el **motor de propagación** del virus. La lógica condicional de tocar la casa es la misma arquitectura lógica de una **infección viral computacional**.

Verificación de Sistemas de la Sesión

Comprobación de Bloques

¿Pudiste fortalecer el uso de diferentes tipos de bloques (Movimiento, Eventos, Sonido) en el ecosistema de Scratch?



Sí



Parcial



Aún no

Comprensión de Secuencia

¿Comprendes firmemente por qué el orden estricto de la secuencia de instrucciones es vital para el éxito de la programación?



Sí



Parcial



Aún no



Si tu estado es 'Parcial' o 'Aún no', revisa la matriz de bloques (diapositiva 11) y discute el flujo lógico con tu equipo técnico y tu docente antes de avanzar al desarrollo del virus.

Siguientes Pasos: Protocolo P-P-I

PENSAR



¿Qué otras dinámicas del mundo real podríamos simular en Scratch usando estas mismas bases lógicas? (Ej: Ecosistemas, tráfico vehicular, sistemas climáticos).

PRESENTAR



Comparte tu visión en plenaria sobre cómo el bloque de movimiento aleatorio mejorará la imprevisibilidad de nuestra simulación del virus.

INTEGRAR



Reúne las ideas de tu equipo para prepararnos. En la próxima sesión, inicializaremos la construcción de nuestro paciente cero.