

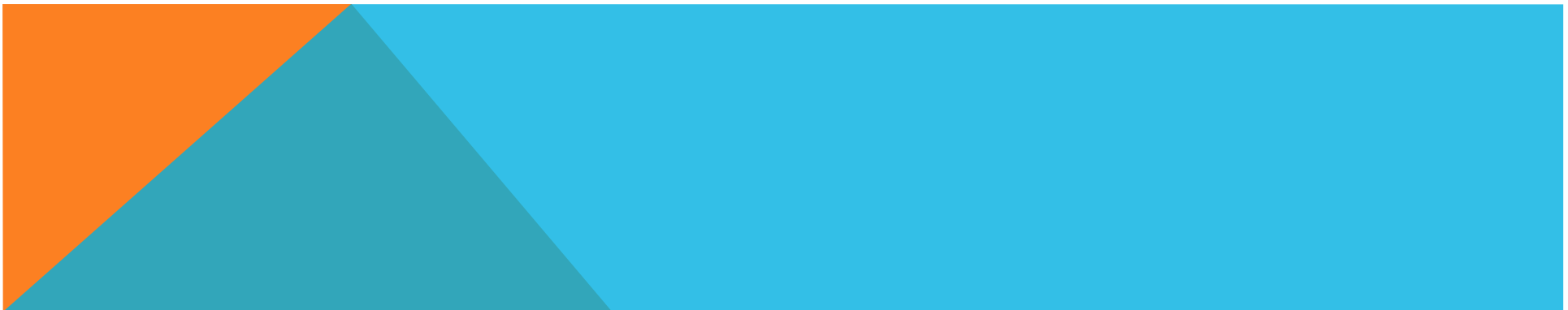
BASES DIDÁCTICAS Y RECURSOS PARA TRABAJAR EL PENSAMIENTO COMPUTACIONAL EN EL AULA

fernandoposada@gmail.com



ÍNDICE DE CONTENIDOS

- Importancia del Pensamiento Computacional (PC)
- Concepto de PC
- El enfoque interdisciplinar del PC
- Orientaciones metodológicas para trabajar PC
- El PC y el pensamiento humano
- Algoritmos y diagramas de flujo
- El PC en el currículum oficial de algunos países
- Recursos para trabajar el PC en el aula
- Robótica educativa



The background features a large orange shape on the right side, a blue triangle on the left, and a teal triangle at the bottom left. The text is centered in the white space between the blue and orange areas.

**IMPORTANCIA DEL
PENSAMIENTO
COMPUTACIONAL**

PENSAMIENTO COMPUTACIONAL, ¿PARA QUÉ?

- Tendencia europea y mundial: incorporar el pensamiento computacional al currículum oficial
- De ciudadanos consumidores de tecnología a ciudadanos productores de tecnología
- Se fomenta el liderazgo, la innovación y el emprendimiento



ALGO MÁS QUE APRENDER A PROGRAMAR

Computational Thinking = resolución de problemas con una metodología propia que incorpora ...

- CREATIVIDAD
- RAZONAMIENTO
- PENSAMIENTO CRÍTICO



LA CREACIÓN DE TECNOLOGÍA COMO FUENTE DE RIQUEZA

- Estimular la formación de un colectivo de creadores digitales que se enfrenten a desafíos complejos, interdisciplinarios, innovadores y de interés público.



The background features a white space with abstract geometric shapes. On the left, there is a light blue triangle pointing right. Below it is a darker teal triangle pointing right. To the right of these is a large orange trapezoidal shape that tapers towards the top right corner.

**CONCEPTO DE PENSAMIENTO
COMPUTACIONAL**

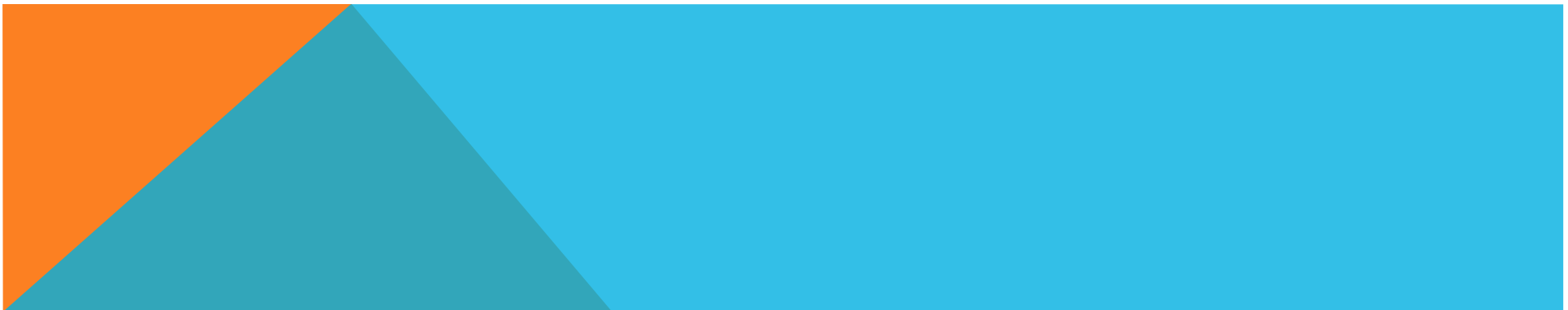
CONCEPTO DE PENSAMIENTO COMPUTACIONAL

Procesos del pensamiento humano implicados en la formulación de problemas y su resolución aplicando un agente de procesamiento de información, bien humano o bien máquina. Jeannete Wing (2010) ⁽¹⁾



HABILIDADES PENSAMIENTO COMPUTACIONAL

1. Formular problemas → adaptarlos a la computación
2. Organizar/analizar datos → patrones y conclusiones
3. Representar datos → descomposición y abstracción
4. Automatizar → diseño creativo de algoritmos
5. Codificar → crear el programa
6. Depurar → resolver los errores
7. Analizar posibles soluciones → eficiencia
8. Generalizar → transferencia de la solución



ACTITUDES PENSAMIENTO COMPUTACIONAL

- Confianza en el manejo de la complejidad
- Persistencia al trabajar los problemas difíciles
- Tolerancia a la ambigüedad
- Habilidad para enfrentarse a problemas
- Habilidad para comunicarse y trabajar con otros



CONCEPTOS COMPUTACIONALES BÁSICOS

1. Abstracciones y generalizaciones
2. Sistemas de símbolos y representación abstracta
3. Noción algorítmica del control de flujo
4. Descomposición estructurada del problema
5. Pensamiento iterativo, recursivo y paralelo
6. Lógica condicional
7. Depuración y detección sistemática de errores
8. Restricciones de eficiencia y desempeño

Grover y Pea (2013) ⁽²⁾

The background features a large orange trapezoidal shape on the right side, and a blue triangular shape on the left side. The blue triangle is composed of two overlapping triangles: a lighter blue one on top and a darker teal one on the bottom. The text is centered within the white space between these shapes.

**EL ENFOQUE INTERDISCIPLINAR
DEL PENSAMIENTO
COMPUTACIONAL**

EL RETO DEL PENSAMIENTO COMPUTACIONAL

- El principal reto para poder incorporarlo al aula es el diseño y puesta en práctica de proyectos en los cuales el pensamiento computacional se integra de una forma transversal a todas las áreas para favorecer un aprendizaje constructivo, conectivo, competencial y tecnológico



EXPERIENCIAS DE APRENDIZAJE INTERDISCIPLINAR

- Computational Thinking Teacher Resources (2016) ⁽³⁾
- 9 ejemplos de experiencias de aprendizaje
- El pensamiento computacional como enfoque transversal y basado en el aprendizaje por proyectos
- Ejemplo: “Cadena alimentaria” Ciencias Nivel K-4 (9-10 años). Crear una animación en Scratch de una cadena alimentaria.



EL PENSAMIENTO COMPUTACIONAL SIN ORDENADORES?

- El pensamiento computacional como sistema de resolución de problemas.
- Tareas con juegos, puzzles, pinturas de colores, dibujos, tarjetas, cartulinas ...
- Ej: Computer Sciences Unplugged csunplugged.org



The background features a white central area with a diagonal split. The bottom-left corner is a light blue triangle, and the bottom-right corner is a teal triangle. The rest of the bottom half is a large orange trapezoidal shape.

**ORIENTACIONES METODOLÓGICAS
PARA TRABAJAR EL
PENSAMIENTO COMPUTACIONAL**

SUGERENCIAS METODOLÓGICAS I

En “Programar para aprender” de Código 21 (2014) ⁽⁴⁾:

- Estimular la creatividad (computación creativa)
- Adaptarse al contexto en las situaciones
- Trabajar la exposición oral y el trabajo en grupo
- Promover reflexión y sentido crítico sobre los productos
- Remezclar y reutilizar respetando derechos de autor



SUGERENCIAS METODOLÓGICAS II

- Trabajar sin ordenador
- Buscar audiencia a los productos del alumnado
- Tener en cuenta la inclusión de género
- Aprovechar las interfaces multilingües
- Evaluar: portfolio propio, entrevista y capacidad de analizar otros proyectos



The background features a white central area. On the left, there is a blue triangle pointing right. Below it is a teal triangle pointing right. On the right, there is a large orange shape that is a rectangle with its top-left corner cut off by a diagonal line.

**EL PENSAMIENTO
COMPUTACIONAL Y EL
PENSAMIENTO HUMANO**

COMPONENTES DEL PENSAMIENTO HUMANO I

Zapata-Ros (2015) ⁽⁵⁾ define para el PC estos componentes:

1. Análisis ascendente: del abstracto al concreto
2. Análisis descendente: del concreto a lo abstracto
3. Pensamiento heurístico: atajos cognitivos eficientes
4. Pensamiento lateral y divergente: posibles soluciones
5. Creatividad



COMPONENTES DEL PENSAMIENTO HUMANO II

6. Resolución de problemas. Incorporando máquinas
7. Pensamiento abstracto. Creación de modelos
8. Recursividad. Ejemplo: cálculo del factorial
9. Iteración. Bucles repetitivos
10. Aproximaciones sucesivas: ensayo y error



COMPONENTES DEL PENSAMIENTO HUMANO III

- 11. Métodos colaborativos. Comunidad de trabajo
- 12. Patrones. Reutilización en situaciones similares
- 13. Sinéctica. Organización de aportaciones colectivas
- 14. Metacognición. Reflexión sobre el propio proceso creativo





ALGORITMOS Y DIAGRAMAS DE FLUJO

ALGORITMO

- Secuencia ordenada de pasos para resolver un problema.
- Ejemplos de documentos instructivos para ...
 - Montar un mueble de IKEA
 - Preparar un plato de comida



ALGORITMOS CON PSEUDOCÓDIGO

- Pseudocódigo = descripción informal de un algoritmo usando un lenguaje a medio camino entre el lenguaje natural y el lenguaje de programación.

Ejemplo:

```
imprimir('La lámpara no funciona');  
si (¿Está enchufada la lámpara?) entonces  
    si (¿Está fundida la bombilla?) entonces  
        imprimir('Cambiar la bombilla');  
    si no  
        imprimir('Comprar nueva lámpara');  
    fin-si  
si no  
    imprimir('Conectar el enchufe');  
fin-si
```



ALGORITMOS CON DIAGRAMAS DE FLUJO

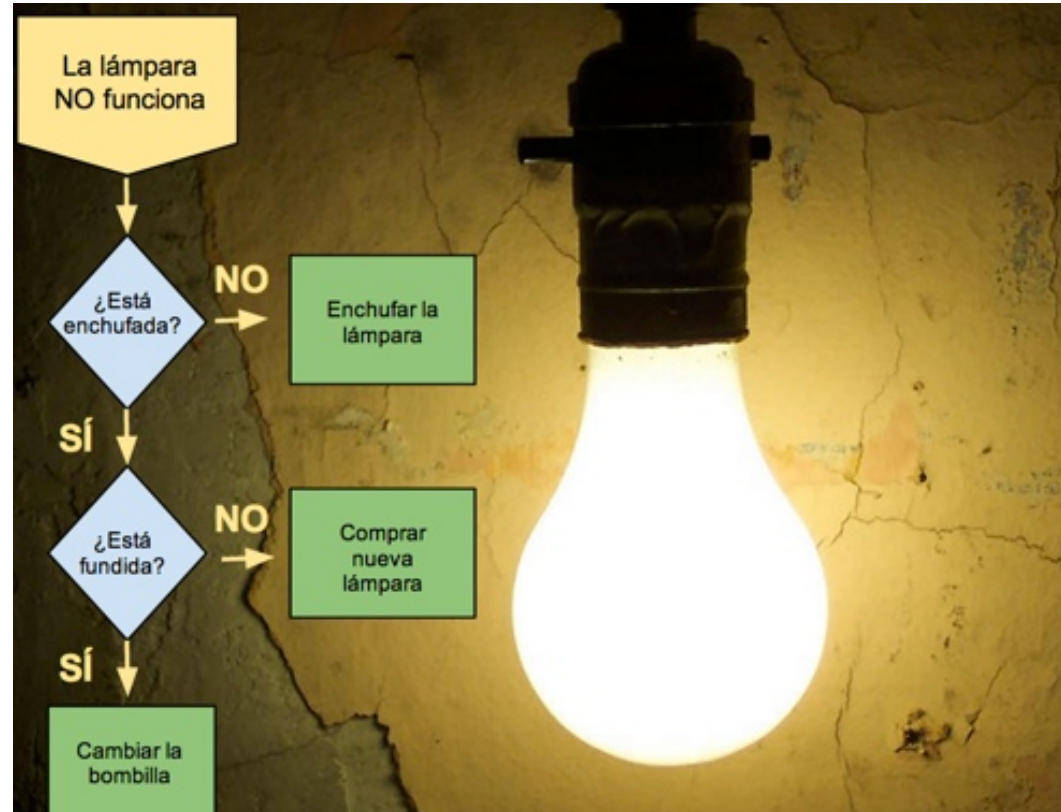



Diagrama de flujo = representación gráfica de un algoritmo o proceso

The background features a large orange trapezoidal shape on the right side, and a blue triangle on the left side. The text is positioned in the white space between these shapes.

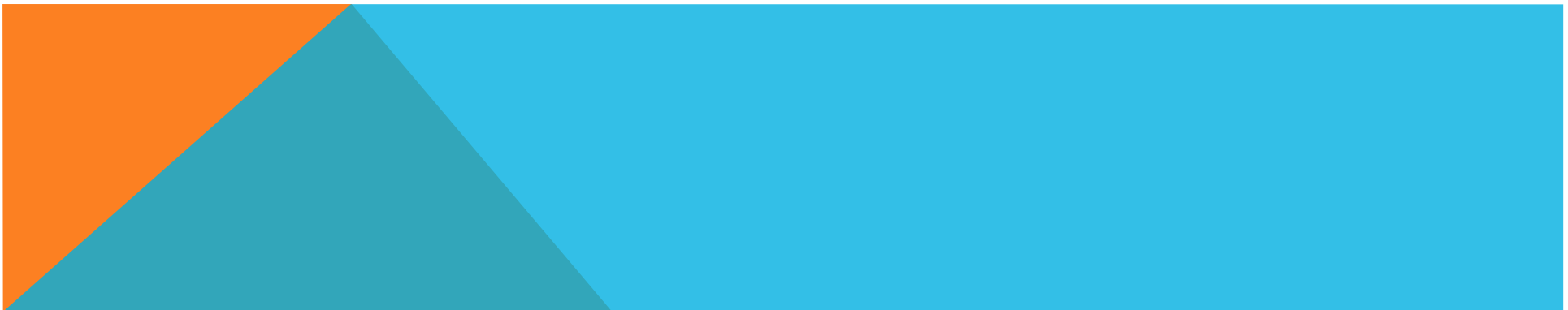
**EL PENSAMIENTO
COMPUTACIONAL EN EL
CURRÍCULUM OFICIAL DE
ALGUNOS PAÍSES**



REINO UNIDO

ASIGNATURA “COMPUTING” EN REINO UNIDO I

- Sustituye a la asignatura ICT = Tecnologías de la Información y la Comunicación
- En Primaria (Key Stage 1 y 2) y en Secundaria (Key Stage (3 y 4)
- Abarca 3 dimensiones:
 - CIENCIAS DE LA COMPUTACIÓN
 - TECNOLOGÍAS DE LA INFORMACIÓN
 - ALFABETIZACIÓN DIGITAL



OBJETIVOS DE “COMPUTING” EN REINO UNIDO

- Conocer y aplicar conceptos de computación.
- Analizar y resolver problemas mediante programación
- Evaluar y aplicar TIC emergentes de forma analítica para resolver problemas
- Ser responsables, competentes, seguros y creativos TIC

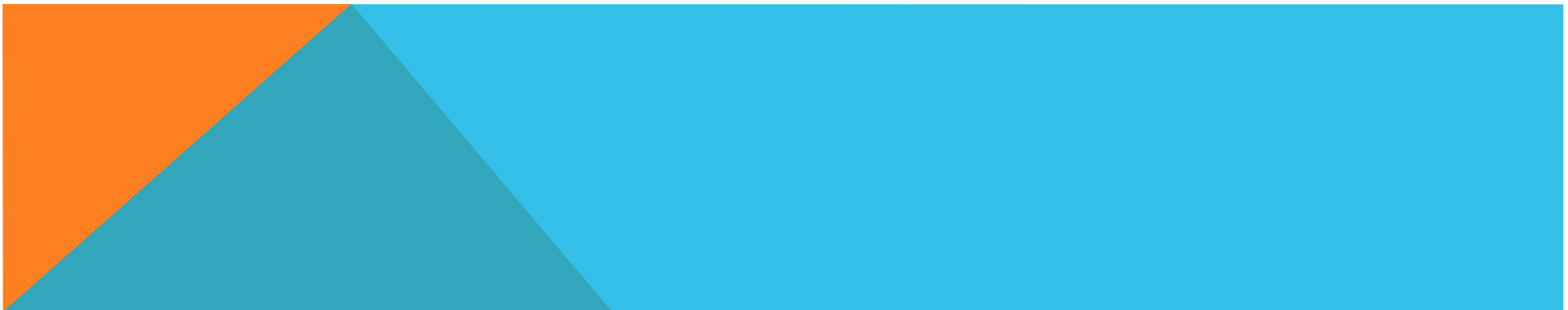


The image features a large, abstract graphic divided diagonally from the bottom-left to the top-right. The upper-left portion is a light blue triangle, and the lower-right portion is a large orange trapezoid. The text "ESTADOS UNIDOS" is written in a bold, black, sans-serif font, oriented diagonally to follow the boundary between the two colors. The text is positioned in the white space between the blue and orange areas.

ESTADOS UNIDOS

MODELO CSTA EN USA I

- CSTA = Computer Science Teachers Association
- Propósito: promover la computación en K-12
- Modelo de estándares K-12 para trabajar Computer Science
- Estándares = descripciones generales y claras de lo que deben aprender los alumnos al finalizar una etapa
- Se especifican estándares por tramos de edades: K-2 (5-8 años), K-5, K-8, K-10 y K-12 (16-18 años)



MODELO CSTA EN USA II

Los estándares se organizan en 5 ejes conceptuales:

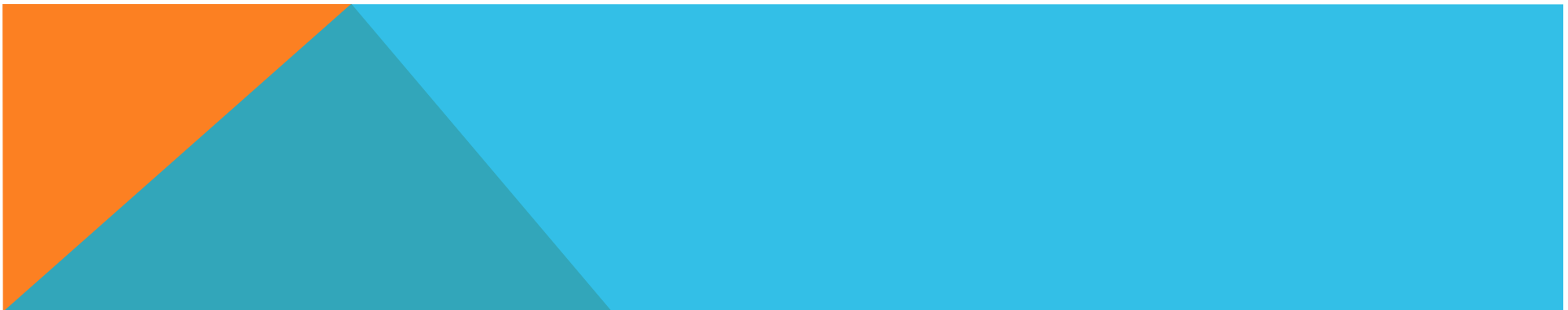
- Sistemas de computación
- Redes e Internet
- Algoritmo y programación
- Datos y análisis
- Impacto de la computación



MODELO CSTA EN USA III

Los estándares se clasifican en 7 ejes prácticos:

- Fomento de la cultura computacional inclusiva
- Colaboración
- Reconocimiento y definición de problemas.
- Desarrollo y uso de abstracciones
- Creación de artefactos computacionales
- Testeo y depuración
- Comunicación sobre computación



MODELO CSTA EN USA IV

Ejemplo:

- Edad: K-2
- Estándar: Analizar y depurar un algoritmo que incluye secuencias y bucles sencillos, con y sin dispositivo computacional.
- Concepto: Algoritmos y programas
- Práctica: Testeo y depuración



COMUNIDAD DE MADRID

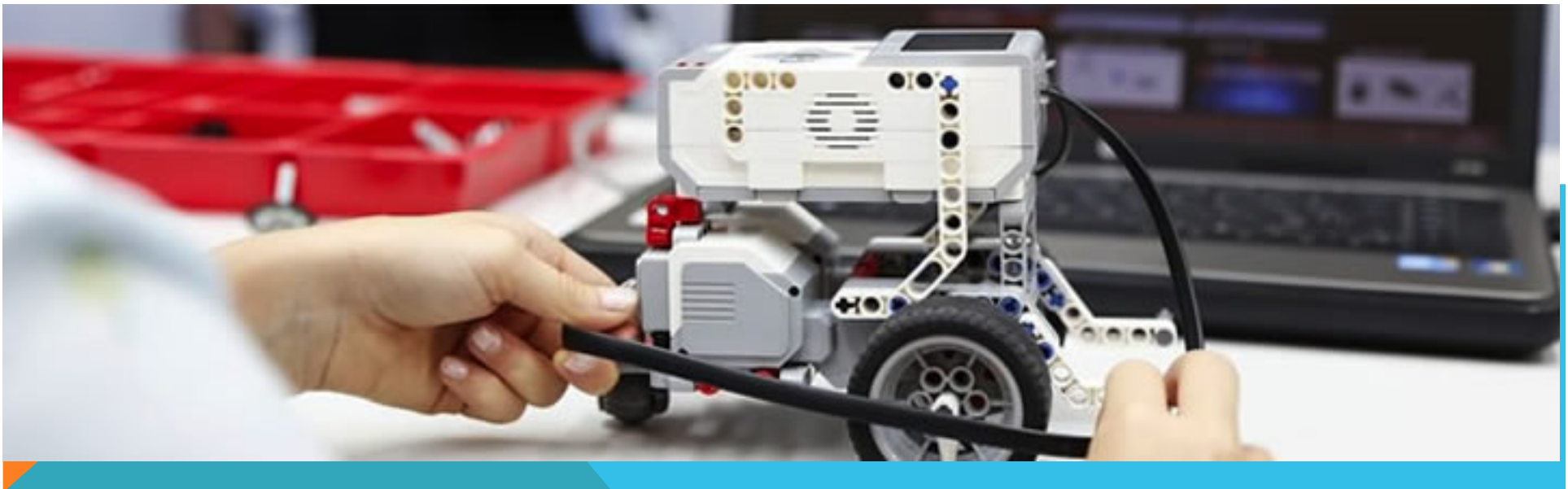


ASIGNATURA “TECNOLOGÍA, PROGRAMACIÓN Y ROBÓTICA”

Obligatoria en 1º-2º-3º ESO. 2 horas semanales

Los contenidos se agrupan en 5 temas:

- Programación y pensamiento computacional
- Robótica y la conexión con el mundo real
- Tecnología y desarrollo del aprendizaje basado en proyectos
- Internet y su uso seguro y responsable
- Técnicas de diseño e impresión 3D



AL FINALIZAR EL CICLO ...

El alumno/a deberá dominar las siguientes tareas:

- Crear una página web con un gestor de contenidos
- Elaborar una app móvil con MIT App Inventor
- Diseñar un videojuego con Scratch
- Elaborar un programa con lenguaje programación textual
- Manejar una impresora 3D
- Tener conocimientos de robótica Arduino





NAVARRA

The image features a stylized map of Navarre, a region in northern Spain. The map is composed of several geometric shapes. On the left side, there is a light blue triangle. Below it, a darker teal triangle shares the same left edge. The rest of the map is a large orange shape that starts from the right edge of the teal triangle and extends to the right edge of the image. The word "NAVARRA" is written in a bold, black, sans-serif font, rotated diagonally to follow the boundary between the blue and orange areas.

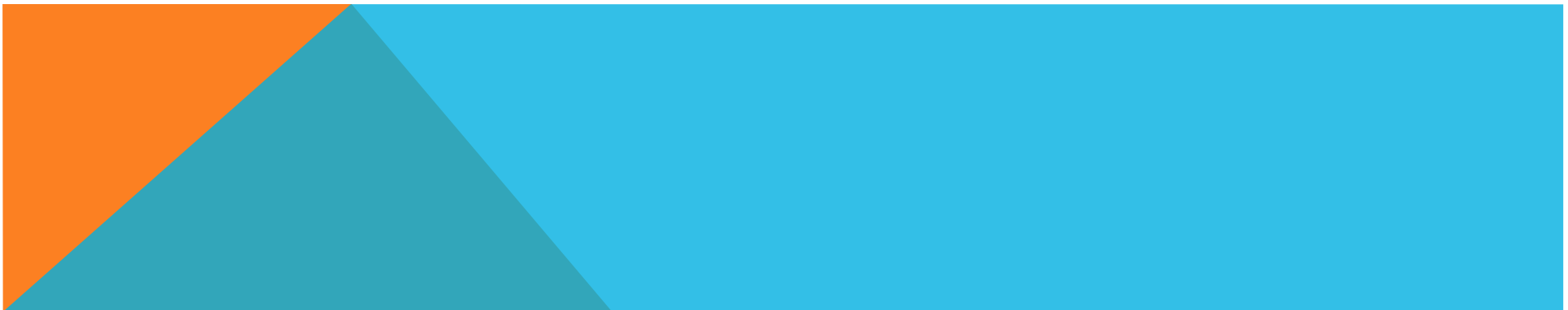
PC EN MATEMÁTICAS DE PRIMARIA I

- En julio 2014 se introduce el trabajo en pensamiento computacional en el currículum oficial del área de MATEMÁTICAS de 4° y 5° de Primaria
- Bloque 1. Procesos, métodos y actitudes
- Contenidos: integración de las TIC, así como de los lenguajes y herramientas de programación en el proceso de aprendizaje
- Criterios de evaluación: utilizar herramientas y lenguajes de programación para modelizar y resolver problemas



ESTÁNDARES DE APRENDIZAJE

- **4° Primaria:** “De manera guiada, realiza un proyecto de programación donde hay que describir el algoritmo, descomponer el problema en partes pequeñas y codificarlo con un lenguaje de programación visual formal (como Scratch)”
- **5° Primaria:** “Diseña y realiza proyectos de programación donde se utilizan secuencias de comandos, bucles, condicionales, variables, así como distintas formas de entrada y salida de datos (interacción con el ordenador)”

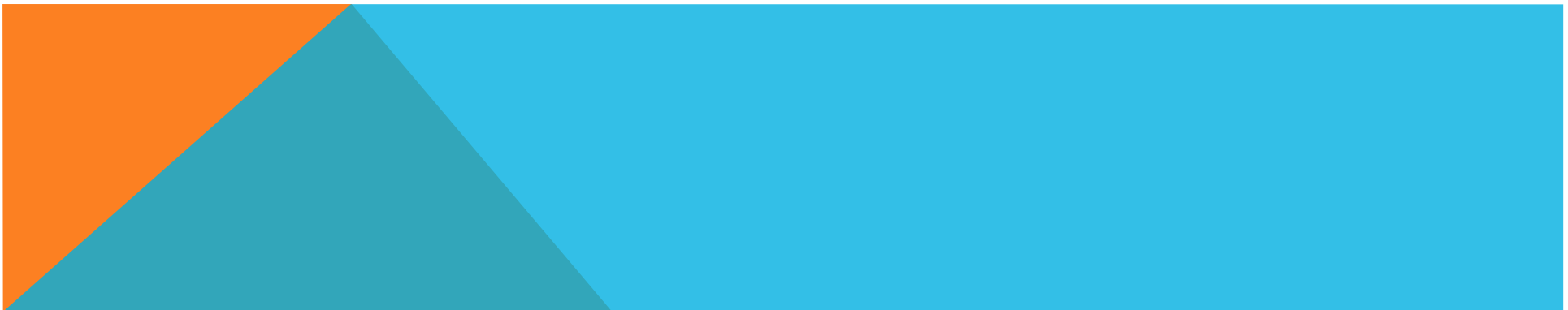


The background features a large orange trapezoidal shape on the right side, and a blue triangle on the left side. The blue triangle is divided into two parts by a diagonal line, with a darker teal section at the bottom-left corner.

**IDEAS SOBRE PENSAMIENTO
COMPUTACIONAL Y CURRÍCULUM**

IDEAS SOBRE PC Y CURRÍCULUM

- El Pensamiento Computacional amplía el concepto de Competencia Digital
- Propone una metodología de resolución de problemas que incluye razonamiento, creatividad y pensamiento crítico
- Puede integrarse en el currículum de forma transversal o bien como un área específica
- Requiere diseñar una propuesta progresiva y adaptada a lo largo de la escolarización obligatoria del alumnado



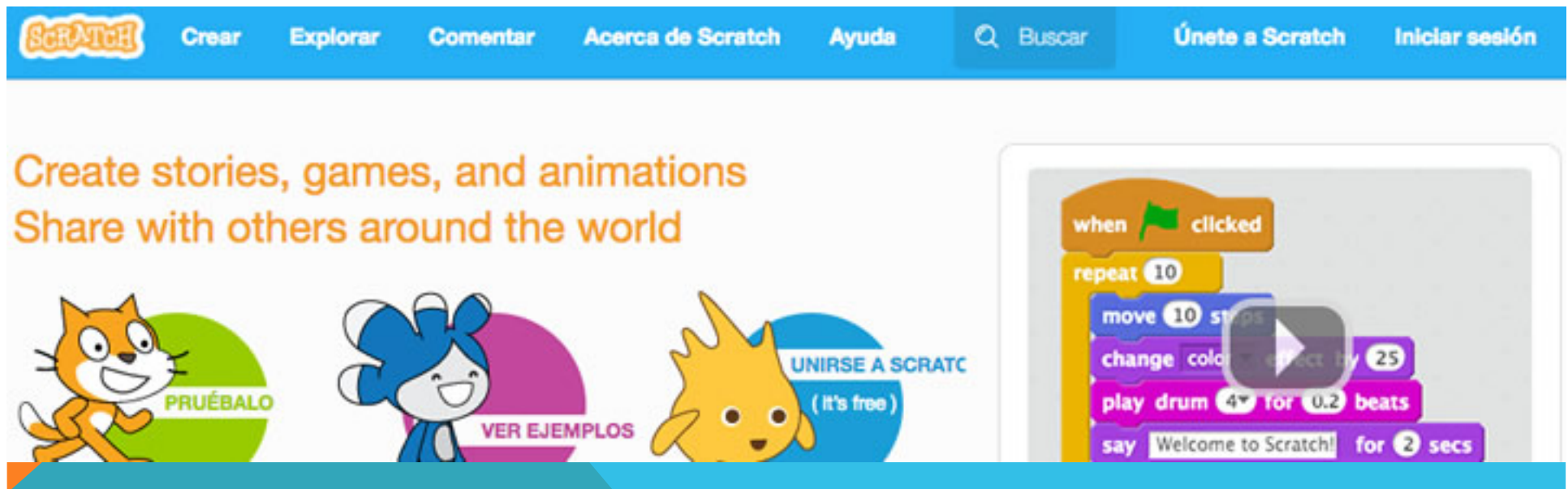


**RECURSOS PARA TRABAJAR
PC EN EL AULA**

SCRATCH

scratch.mit.edu

- Entorno de programación visual pensado para jóvenes
- Permite la introducción al pensamiento computacional y la expresión creativa de ideas mediante la creación de animaciones interactivas.
- Es software gratuito desarrollado por el MIT (Instituto Tecnológico de Massachusetts)



CODE.ORG

code.org


- Fundación internacional para la promoción del pensamiento computacional
- Propone interesantes tutoriales autoguiados en “La hora del código”



Alumnos
Explora todos nuestros tutoriales



Educadores
Enseña a tus alumnos



Hora del Código
Cualquiera puede aprender.
Comienza hoy mismo



Código de Flappy
Crea y comparte tu propio juego
Flappy

Explora Code Studio



SCRATCHJR

www.scratchjr.org

- App Android/iOS para la iniciación a la programación de niños/as de 5-7 años
- Inspirado en Scratch para crear historias y juegos interactivos

ScratchJr

Programación para niños

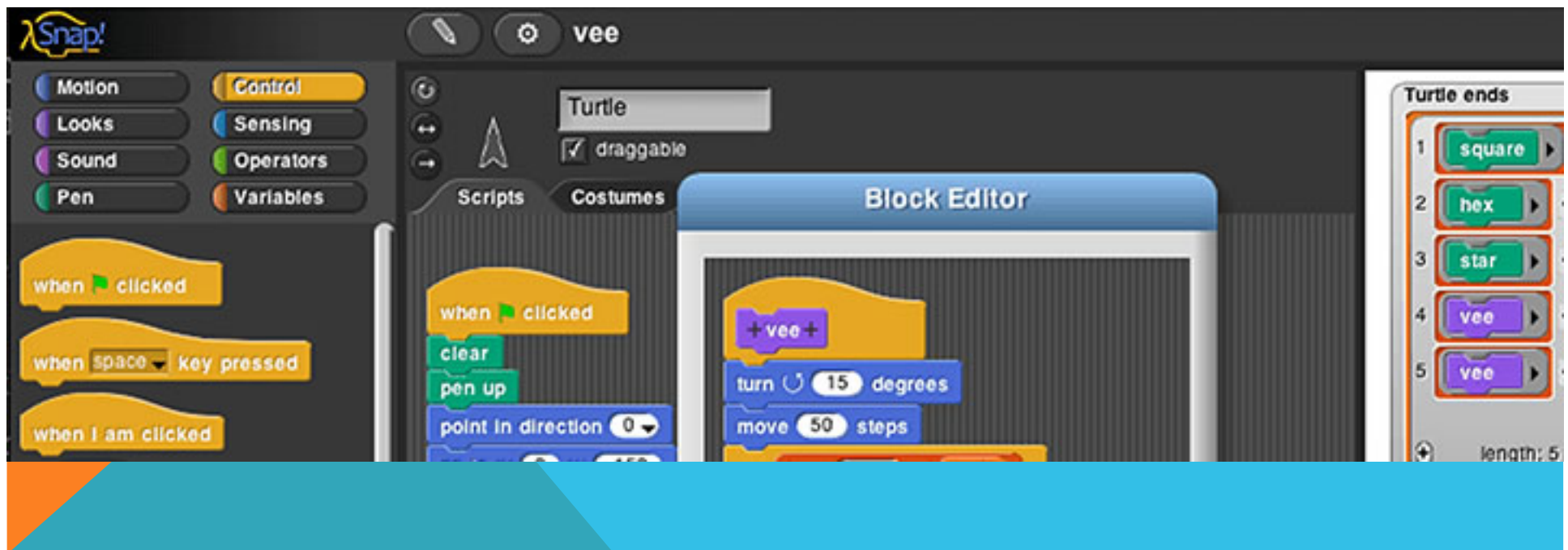
¡La programación es la nueva alfabetización! Con ScratchJr, niños (entre 5 y 7 años) pueden programar sus propias historias y juegos interactivos. En el proceso, aprenden a resolver problemas,



SNAP!

snap.berkeley.edu

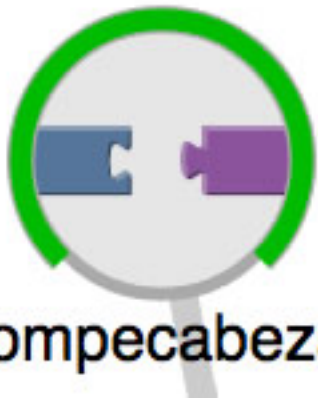
- Implementación extendida de Scratch
- Permite construir nuevos bloques (procedimientos)
- Ideal para alumnado de Secundaria y Universidad



BLOCKLY GAMES

blockly-games.appspot.com

- Recopilación de juegos para desarrollar el pensamiento computacional
- Más adecuado para alumnado de Primaria



Rompecabezas

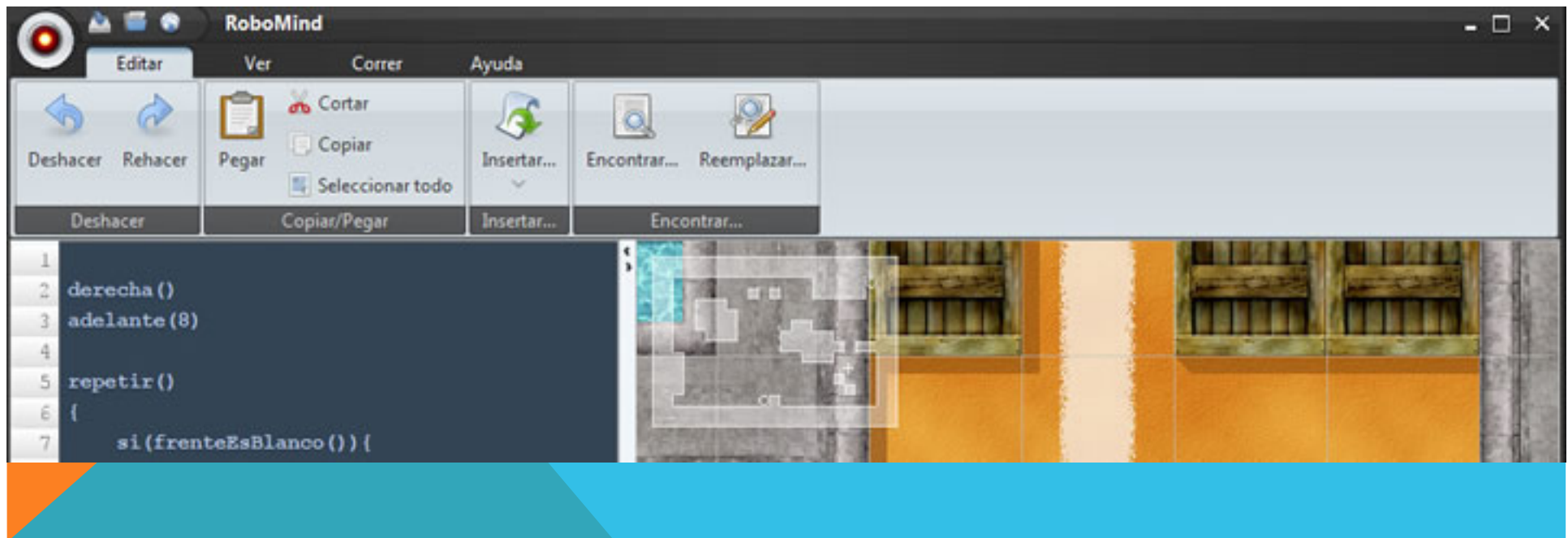


Juegos para los programadores del futuro. [Más informa](#)

ROBOMIND

www.robomind.net/es

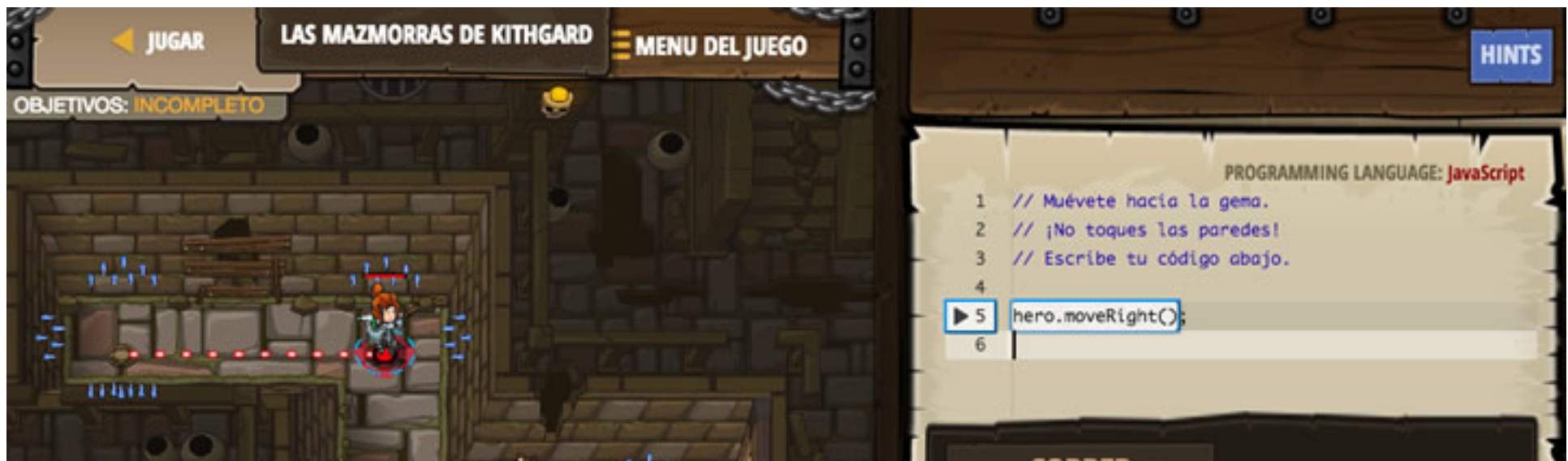
- Lenguaje de programación textual donde se programa el comportamiento de un robot.
- Dispone de un amplio repertorio de acciones, bucles, condicionales, procedimientos, etc.
- Ideal para Secundaria y Universidad



CODE COMBAT

codecombat.com

- Juego de aventuras basado en programación textual
- Se trata de programar el movimiento del avatar por los distintos escenarios
- Para Primaria y Secundaria



TYNKER

www.tynker.com

- Editor en línea o bien app para Android/iOS
- Basado en programación visual para el diseño de juegos y aplicaciones.
- Para alumnado a partir de 8 años



FREE PLAY

PARENTS

EDUCATORS

SIGN IN

JOIN FOR FREE

FREE CODING GAMES

GRADE K+

GRADE 3+

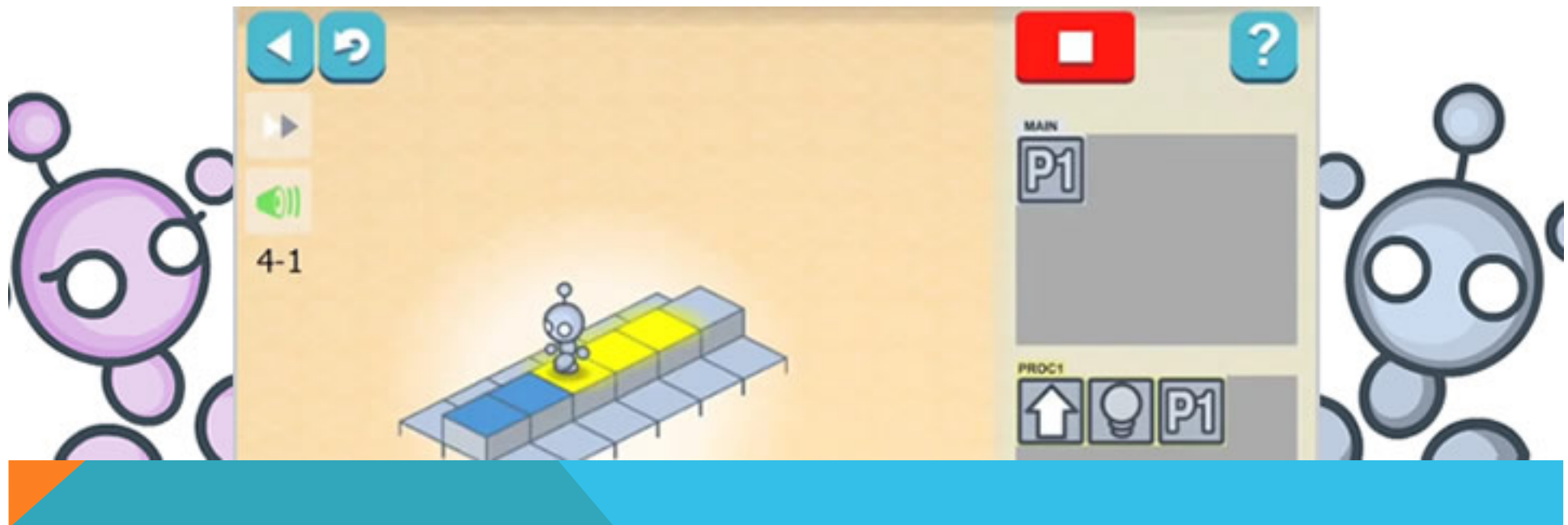
GRADE 6+

ALL GRADES

LIGHTBOT: UNA HORA DE CÓDIGO

lightbot.com

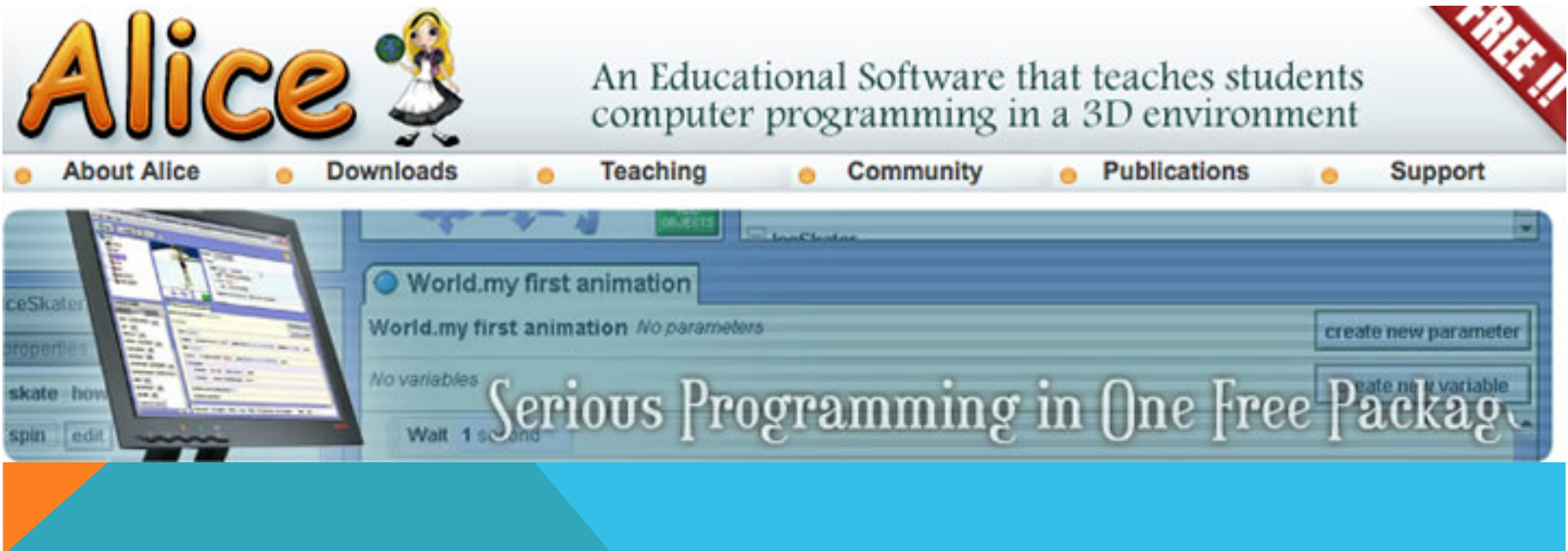
- Aplicación en línea o bien App para Android/iOS
- Juego de programación visual de 20 niveles donde se resuelven distintas situaciones problema mediante la conexión de una secuencia de bloques.
- Para Primaria y Secundaria




ALICE

www.alice.org

- Software gratuito que se instala en el ordenador
- Está orientado a la creación de animaciones interactivas, narración de historias, etc
- Para alumnado de Bachillerato y Universidad



The image shows a promotional banner for Alice software. On the left, the word "Alice" is written in a large, orange, bubbly font next to a small cartoon character of Alice. To the right, the text reads "An Educational Software that teaches students computer programming in a 3D environment". Below this is a navigation bar with links: "About Alice", "Downloads", "Teaching", "Community", "Publications", and "Support". A red diagonal banner in the top right corner says "FREE!!". Below the navigation bar is a screenshot of the Alice software interface, showing a 3D scene with a character and various programming controls. The text "Serious Programming in One Free Package" is overlaid on the screenshot. The bottom of the banner has a blue and orange geometric design.

Alice  An Educational Software that teaches students computer programming in a 3D environment

● About Alice ● Downloads ● Teaching ● Community ● Publications ● Support

World.my first animation
World.my first animation No parameters create new parameter
No variables create new variable
Wait 1 second

Serious Programming in One Free Package

KODABLE

www.kodable.com

- Propone una secuencia progresiva de ejercicios para trabajar conceptos computacionales en función de la edad
- Tiene versión web y versión app para móviles.
- Para Primaria y Secundaria



The image shows a promotional banner for Kodable. On the left, the Kodable logo is displayed in yellow, with links for 'Learn More' and 'Teacher Resources'. The main text reads 'Kids Can Code. And you can help them.' followed by 'Programming Curriculum for Elementary' and 'Used by 1 in 3 US Elementary Schools'. On the right, there are two buttons: 'Student Login' (orange) and 'Parent/Teacher Login' (green). The background features a classroom setting with a computer monitor and a large screen displaying a coding interface.

Kodable Learn More Teacher Resources

Student Login Parent/Teacher Login

Kids Can Code.
And you can help them.

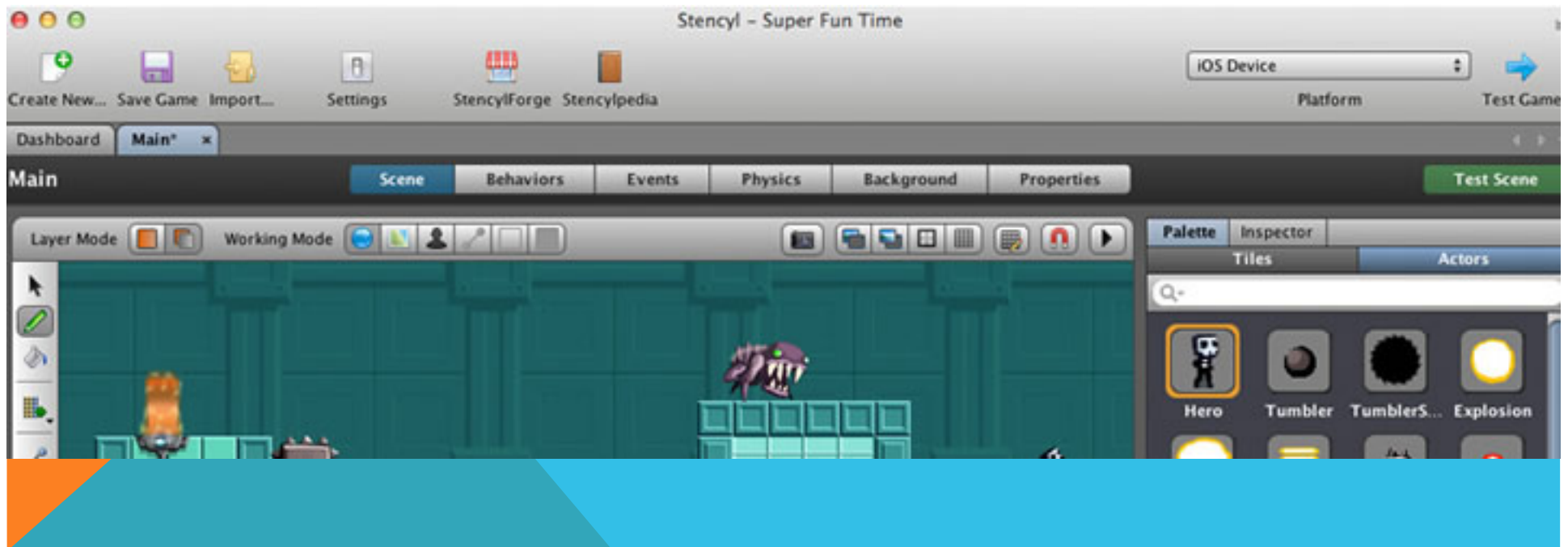
Programming Curriculum for Elementary

Used by 1 in 3 US Elementary Schools

STENCYL

www.stencyl.com

- Software multiplataforma instalable para crear juegos mediante programación visual
- Para Secundaria



CODEHUNT

www.codehunt.com

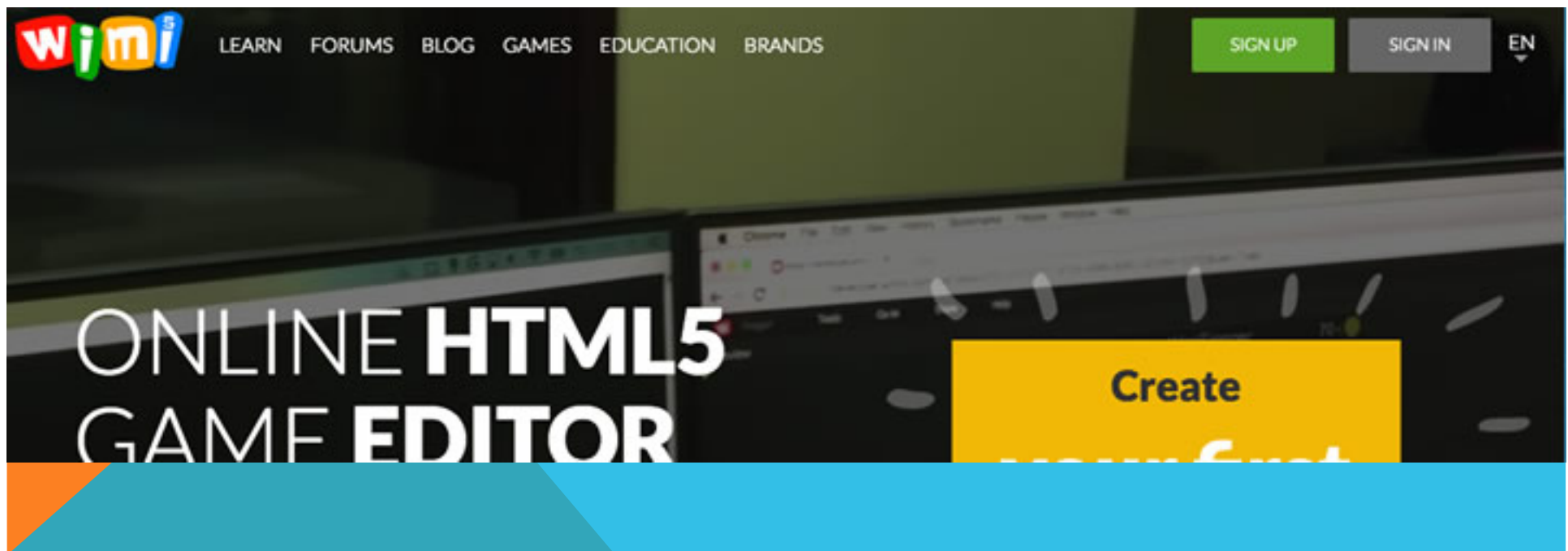
- Juego en línea para aprender a programar
- El cazador de código acumula puntos al descubrir el fragmento de código perdido
- Para Bachillerato y Universidad



WIMI5

www.wimi5.com

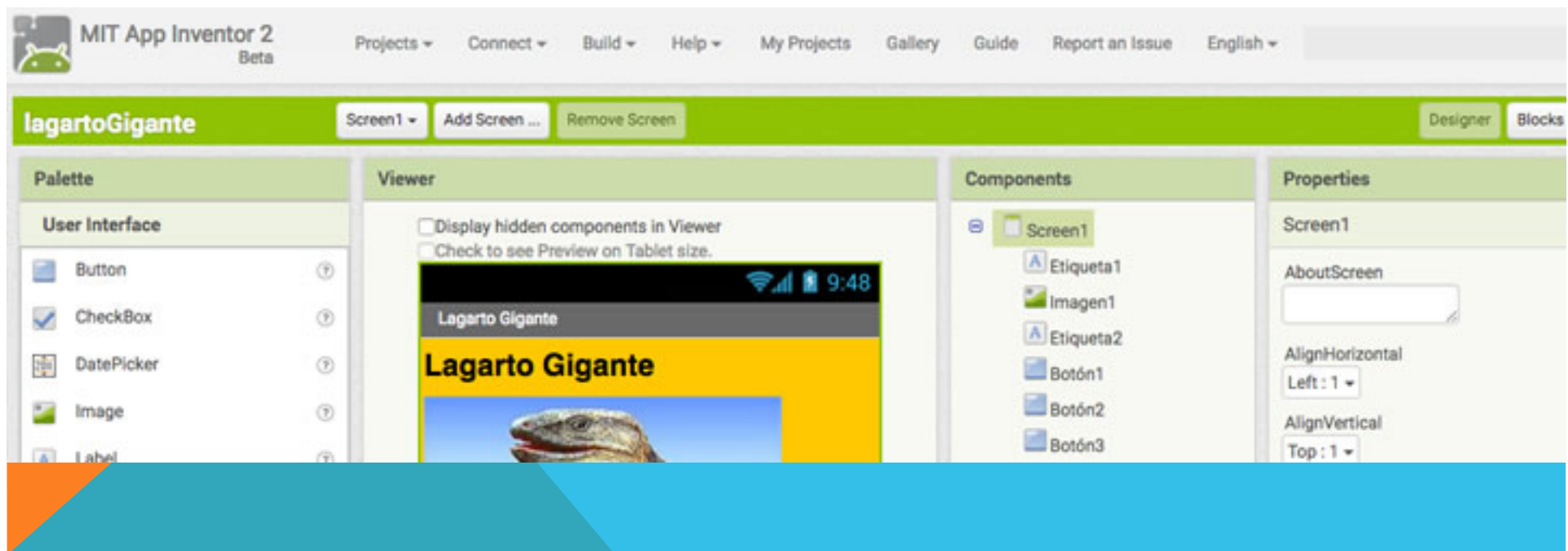
- Editor en línea gratuito para crear y publicar videojuegos en formato HTML5
- La lógica del juego se construye conectando bloques funcionales.
- Para Bachillerato y Universidad



MIT APP INVENTOR 2

ai2.appinventor.mit.edu

- Herramienta gratuita y basada en programación visual para el diseño de apps para dispositivos móviles Android
- Creada por el MIT al igual que Scratch
- Para Primaria, Secundaria y Universidad



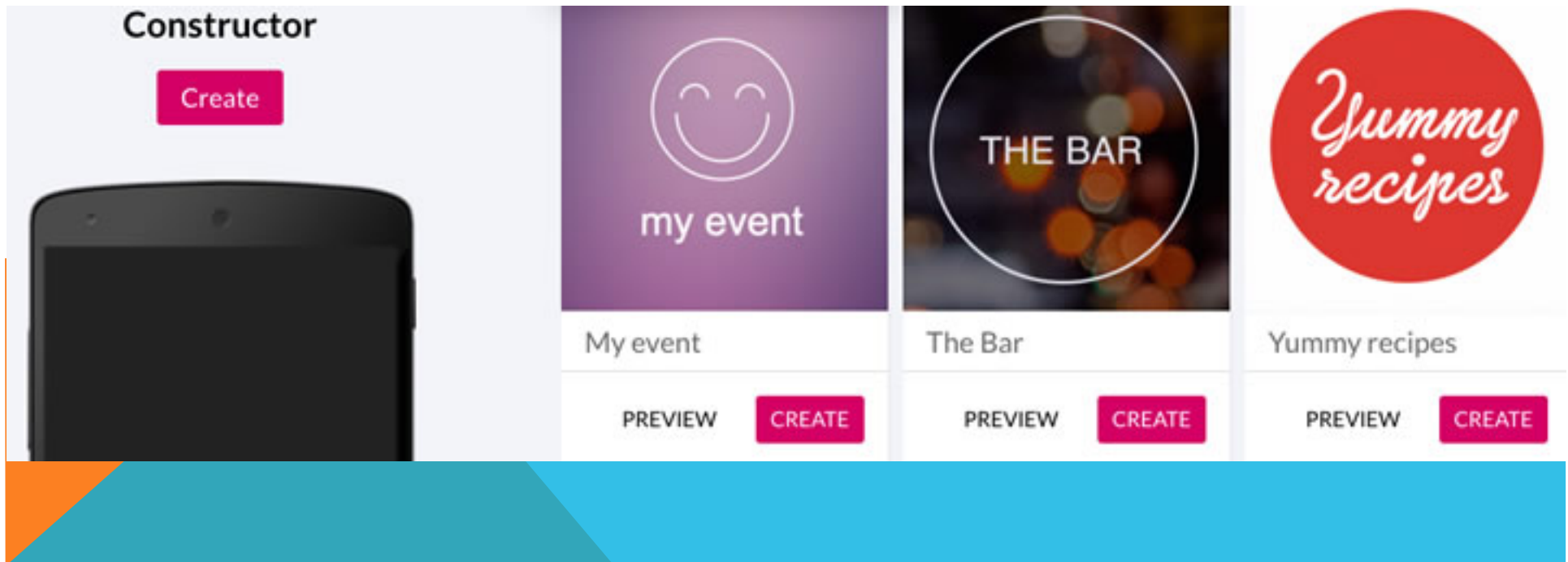
ALTERNATIVAS A MIT APP INVENTOR

- **Appypie:** www.appypie.com
- **App Mark:** www.appmakr.com
- **Mobincube:** www.mobincube.com
- **App Lab:** code.org/educate/applab



DISEÑO DE APPS PARA DISPOSITIVOS MÓVILES

- Más centradas en el diseño que en la programación
- Proceso creativo basado en la clonación de plantillas
- Producto final con un acabado más profesional
- Gratuitas con prestaciones limitadas y publicidad
- De pago por suscripción anual

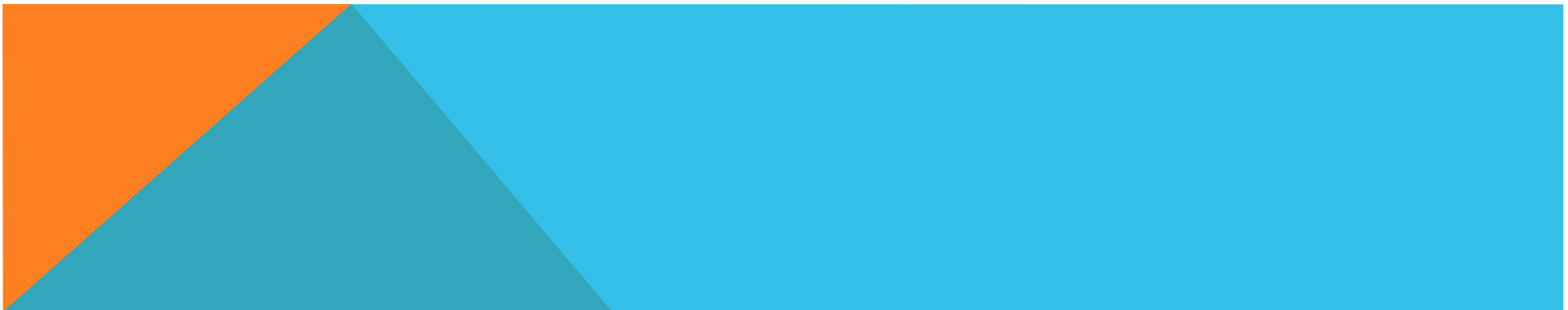




**ALGUNAS IDEAS SOBRE
RECURSOS PARA TRABAJAR PC
EN EL AULA**

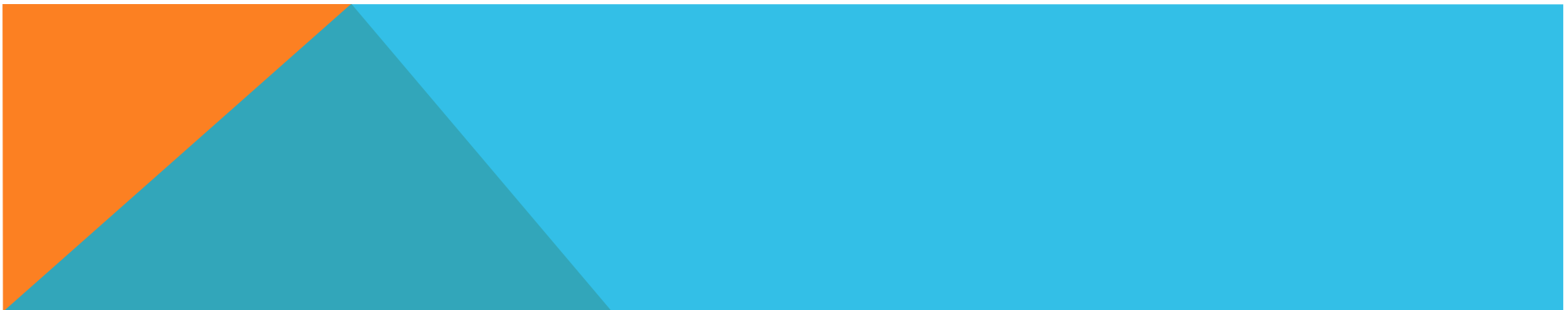
ALGUNAS IDEAS SOBRE RECURSOS PARA TRABAJAR PC EN EL AULA - 1

- Scratch se ha convertido en un estándar para trabajar el pensamiento computacional en el aula
- La programación visual mediante conexión de bloques facilita su aplicación en todas las edades
- En el currículo de Secundaria y Bachillerato se sugiere también el uso de herramientas de programación textual
- Existen múltiples aplicaciones y recursos para diseñar propuestas variadas, motivantes y de dificultad creciente a lo largo de la enseñanza obligatoria



ALGUNAS IDEAS SOBRE RECURSOS PARA TRABAJAR PC EN EL AULA - 2

- MIT App Inventor 2 está basado en programación visual como su hermano pequeño Scratch
- Esta herramienta permite **que los alumnos/as pasen de ser consumidores de apps a ser productores de apps ampliándose las posibilidades del apr.constructivo**
- Las alternativas a MIT App Inventor no demandan tanto PC al estar basadas en la clonación de plantillas aunque su acabado es más profesional



The background features a large orange trapezoidal shape on the right side, and a blue triangle on the left side. The blue triangle is divided into two sections by a diagonal line, with a darker teal section at the bottom-left corner.

ROBÓTICA EDUCATIVA

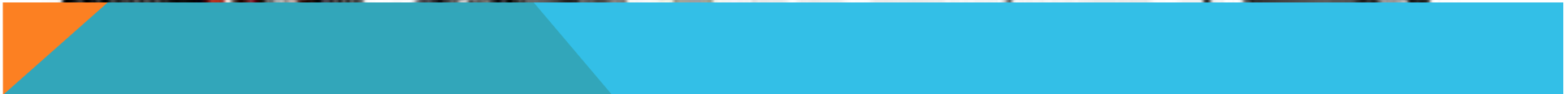
ROBÓTICA EDUCATIVA

- La robótica en el aula es una forma de trabajar el pensamiento computacional
- Se programa el comportamiento de un sistema físico
- Se sobrepasan los límites de la pantalla digital
- De atractivo para el alumnado participante



¿QUÉ ES UN ROBOT?

- Es un sistema eléctrico y mecánico que dispone de una unidad de control donde mediante programación se pueden ejecutar distintos movimientos con intención de realizar una tarea



6 SISTEMAS DE UN ROBOT

1. Chasis o esqueleto: estructura
2. Mecanismos móviles y motor: movimiento
3. Sensores: percepción
4. Unidad de control: procesamiento
5. Conectores: comunicación
6. Fuente de alimentación: energía



PROGRAMACIÓN VISUAL

- Casi todas las soluciones de robótica educativa proponen una interfaz visual de conexión de bloques para diseñar la programación
- El programa se crea directamente en el robot o en un ordenador/tableta y se le envía por wifi, bluetooth o cable usb

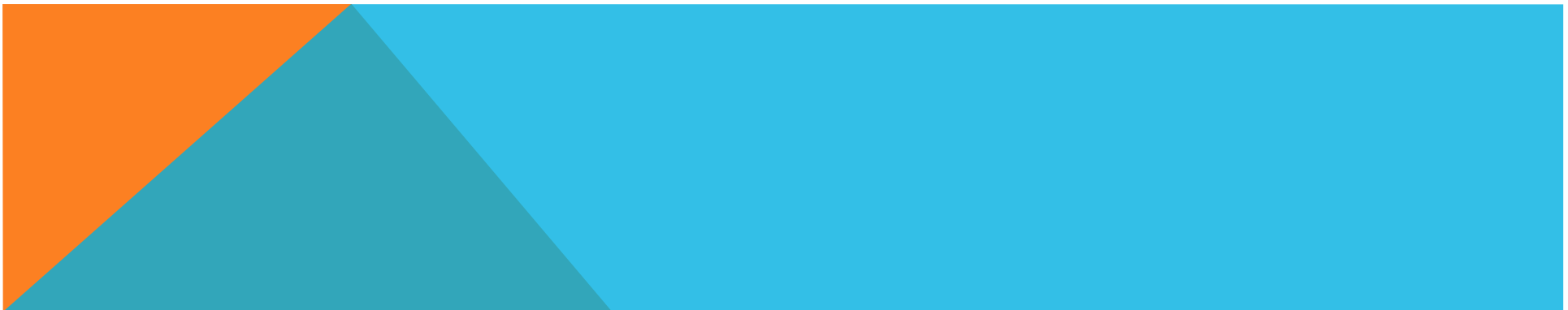


The background features a large orange trapezoidal shape on the right side, and a blue triangle on the left side. The blue triangle is divided into two sections by a diagonal line, with a darker teal section at the bottom-left corner.

**SOLUCIONES MÁS POPULARES
DE ROBÓTICA EDUCATIVA**

LEGO WEDO 2

- Pack que incorpora la conexión por Bluetooth al ordenador
- Kit de 280 piezas para crear 16 modelos de robots
- Electrónica: sensor de distancia y de inclinación, motor y hub bluetooth
- Se programa con herramienta propia y con Scratch
- Edad recomendada: 7-12 años



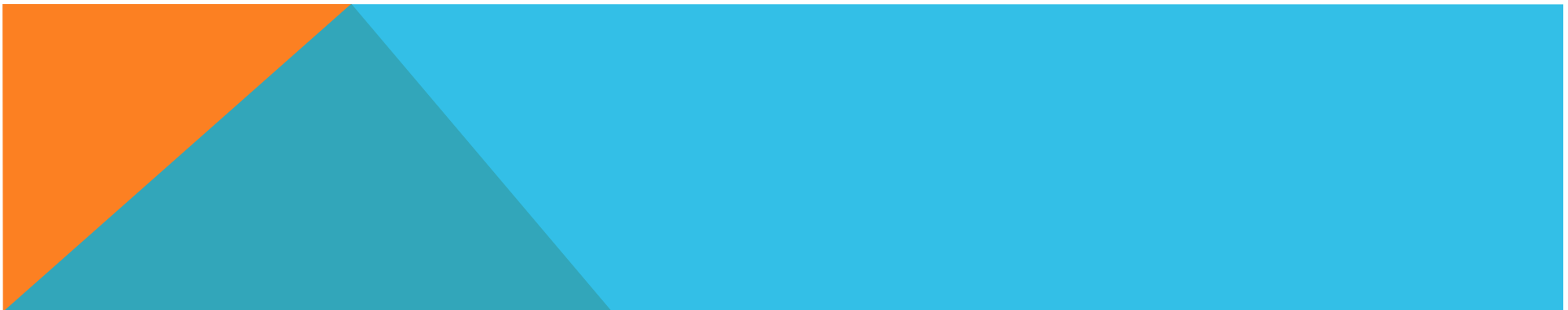
LEGO MINDSTORMS EV3

- Segunda generación de Lego NXT
- Kit con múltiples piezas para crear distintos modelos
- Electrónica: brick inteligente, dos servomotores, un motor mediano, sensor ultrasónica de distancia y presencia, sensor de giros, sensor detector de color y luminosidad y sensores táctiles
- Herramienta propia de programación
- Edad recomendada: 10-21 años



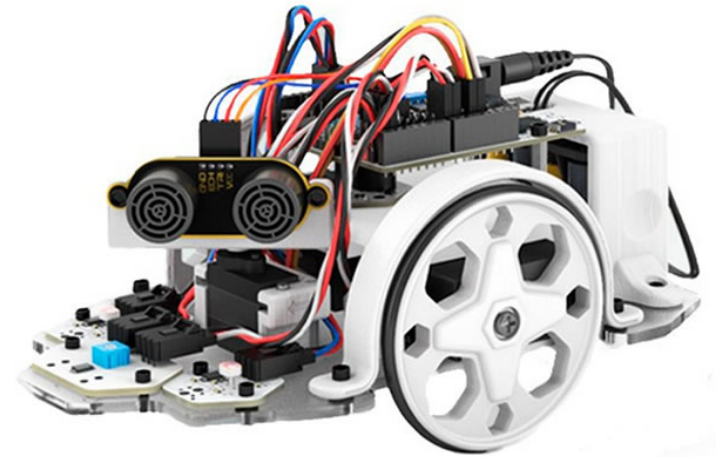
MOWAY BASADO EN ARDUINO

- Robot Arduino autónomo y programable. Se ofrece ya ensamblado para centrarse en la programación
- Electrónica: placa Arduino ATmega 32u4, sensores infrarrojos anticollisión, sensor de intensidad de luz, acelerómetros, micrófono, leds y RF
- Se programa con herramienta propia y con Scratch
- Edad recomendada: de 8 a 18 años



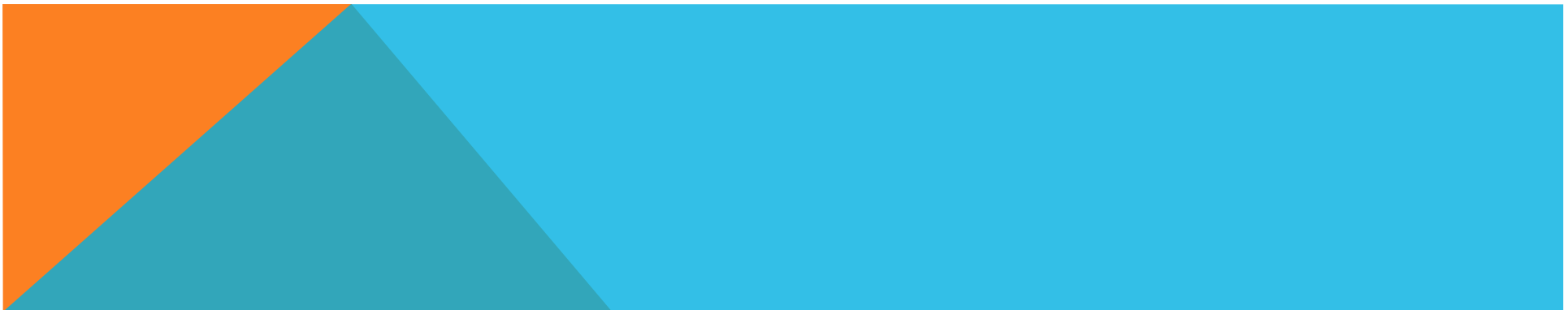
BQ PRINBOT EVOLUTION

- Robot Arduino autónomo y programable
- Se monta fácilmente y admite skins
- Electrónica: placa BQ Zum Core, 2 sensores de luz, un zumbador, un infrarrojo doble, un sensor de ultrasonidos, 1 miniservo y 2 servos de rotación continua
- Se programa con herramienta propia: BitBloq
- Edad recomendada: de 10 a 18 años



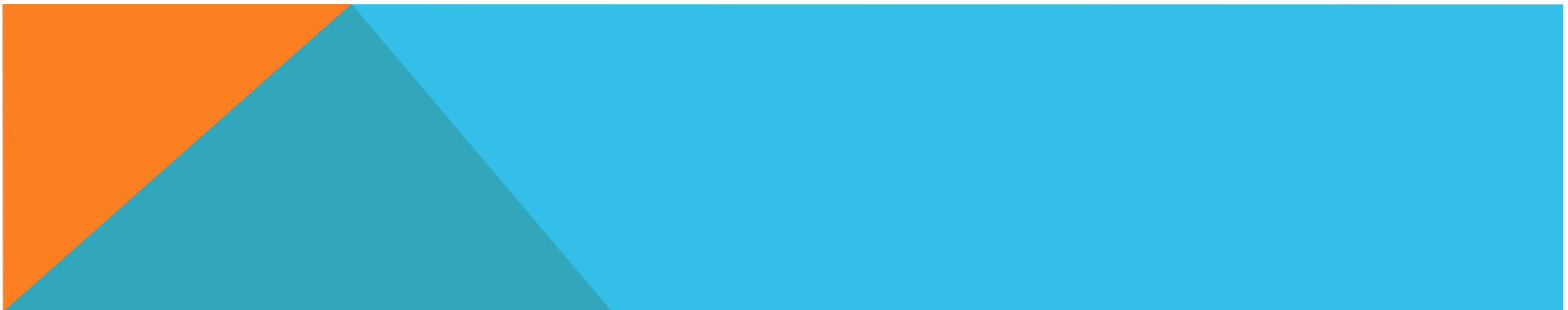
MBOT ROBOT EDUCATIVO

- Robot Arduino autónomo y programable
- Se monta fácilmente y no varía su aspecto
- Mejor modelo 2.4G y no BT para el aula
- Electrónica: placa mCore basada en Arduino UNO, sensor de luz, emisor-receptor de infrarrojos, sensor de proximidad, sensor siguelíneas, módulo BT y 3D, zumbador, leds y pulsador programable
- Se programa con herramienta propia: mBot basada en Scratch
- Edad recomendada: de 8 a 18 años



BEEBOT Y BLUEBOT

- Robot abeja o escarabajo montado para jugar
- Bluebot dispone de módulo BT para conectarse al ordenador o tableta de forma inalámbrica
- Se puede programar directamente sobre el robot o bien desde ordenador o tableta y luego enviárselo
- Material complementario: tapetes, carcasas, etc
- Edad recomendada: 3-7 años



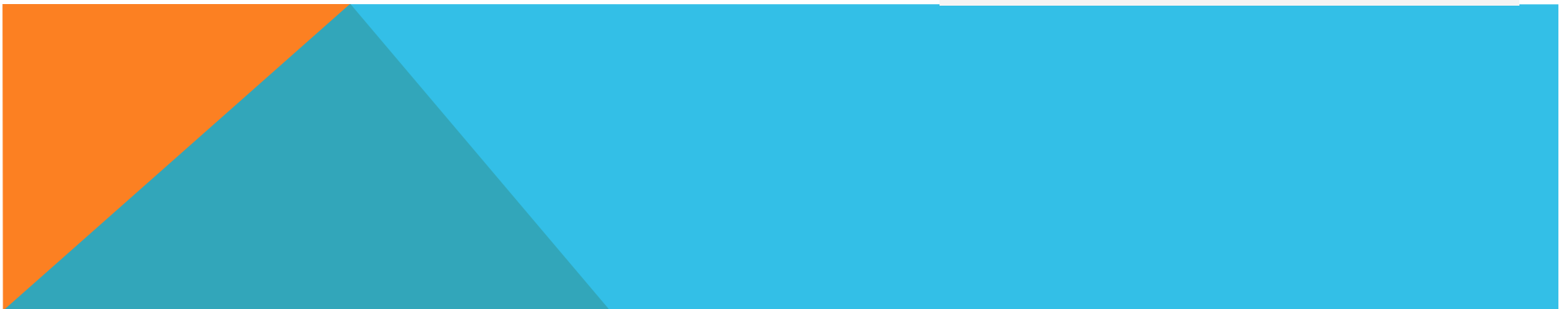
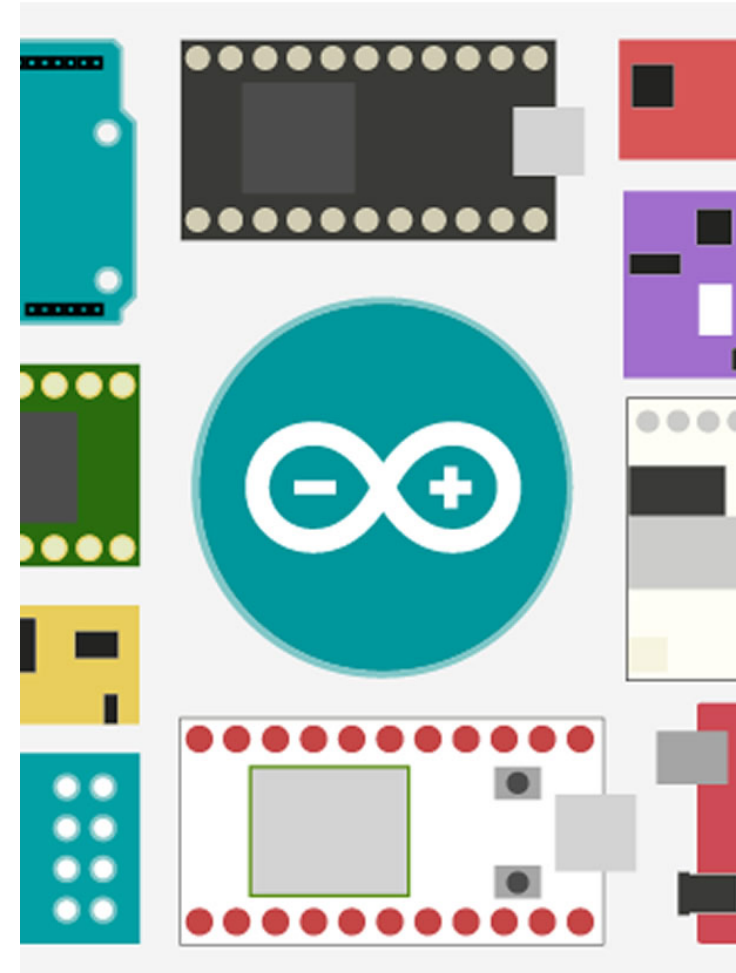
DASH & DOT

- Pareja de robots para iniciarse en la programación visual
- Ofrece distintas apps para controlar y para programar el robot en función de la edad: Wonder, Blockly, Path ...
- Electrónica: 3 micrófonos, 1 altavoz, leds, 4 IR transmisores, 3 detectores de distancia y 2 IR receptores de detección de robot
- Edades recomendadas: 5-12 años



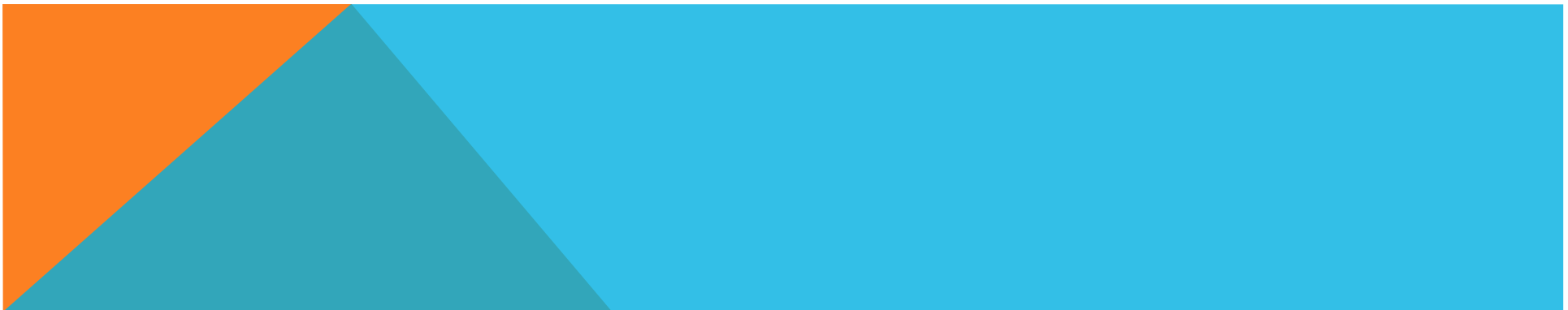
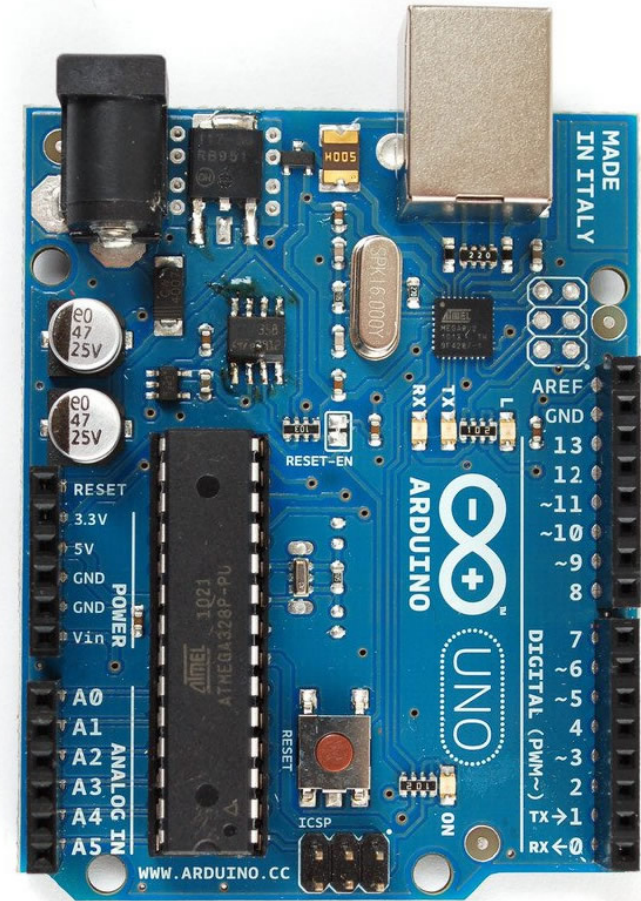
KIT STARTER DE ARDUINO I

- Arduino es una marca de una serie de modelos de placas controladores que permiten la conexión de leds, sensores y motores
- La más popular es el modelo Arduino UNO Rev3



KIT STARTER DE ARDUINO II

- El kit Starter de Arduino contiene una placa de Arduino, un panel de conexiones así como distintos leds, sensores y motores
- Se puede utilizar para programar circuitos electrónicos
- Dispone de un software específico para programación textual
- También se puede programar con S4A, una variante de Scratch para Arduino

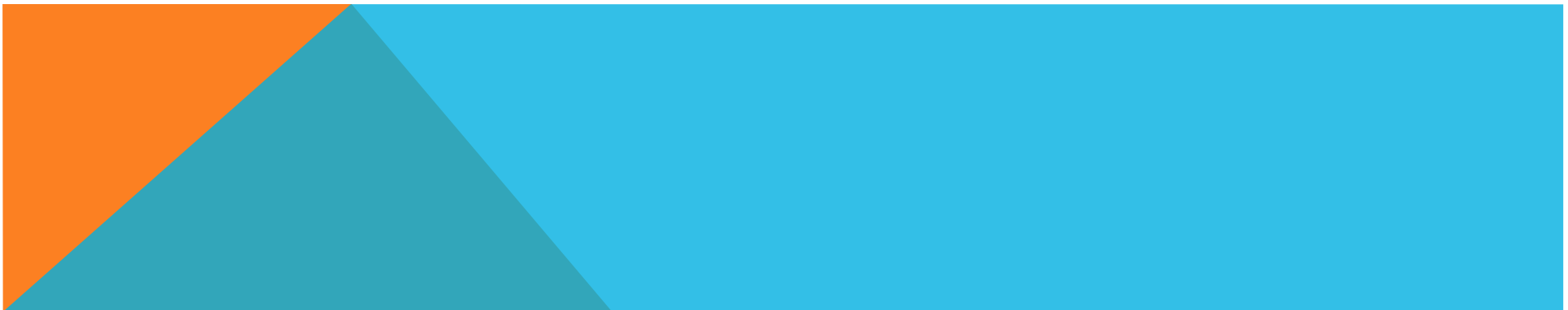


The background features a white central area. On the left, there is a blue triangle pointing right, which overlaps with a teal triangle pointing left. On the right, there is a large orange shape that resembles a trapezoid or a large triangle pointing left, overlapping with the white area.

**ALGUNAS IDEAS SOBRE
ROBÓTICA EDUCATIVA**

ALGUNAS IDEAS SOBRE ROBÓTICA EDUCATIVA

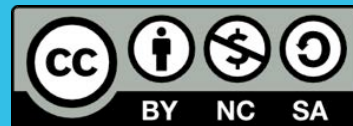
- La robótica es una variante del pensamiento computacional
- El requisito mínimo para que un sistema sea considerado un robot es la programación previa de una secuencia de movimientos. Admite más posibilidades con sensores
- Existen soluciones de robótica basadas en Arduino que por su reducido coste son interesantes para un aula
- Es posible crear una propuesta de robótica escolar variada y progresiva en función de la edad del alumnado usando distintas soluciones



BASES DIDÁCTICAS Y RECURSOS PARA TRABAJAR EL PENSAMIENTO COMPUTACIONAL EN EL AULA

Fotografías obtenidas de freepik.com y flickr.com

Fernando Posada Prieto
fernandoposada@gmail.com



REFERENCIAS

1. Wing, J. 2010. Research Notebook: Computational Thinking - What and Why?
2. Grover y Pea. 2013. Computational Thinking in K-12: A Review of the State of the Field
3. ISTE & CSTA. 2016. Computacional Thinking Teacher Resources. Second Edition
4. Código 21. 2014. Programar para aprender. Orientaciones para el profesorado de Primaria
5. Zapata-Ros. 2015. Pensamiento computacional: una nueva alfabetización digital

