

Reseña de Libro

Coding Games in Scratch: A Step-By-Step Visual Guide to Building Your Own Computer Games

Jon Woodcock

Primera edición, DK Publishing, 2015, 224 páginas.

Este libro tiene como objetivo enseñar el lenguaje de programación Scratch, un lenguaje basado en bloques muy popular creado por el Grupo Lifelong Kindergarten en el Laboratorio de Medios del MIT (Instituto Tecnológico de Massachusetts). Scratch se ofrece de forma gratuita y está enfocado principalmente a personas de 8 a 16 años, pero cualquiera que quiera aprender a programar computadoras lo puede utilizar. Los proyectos que se pueden programar en Scratch son por lo general juegos, animaciones e historias interactivas.

Según su autor, este libro se enfoca más en los niños de 8 a 12 años, con poca o nula experiencia en programación y ofrece una guía visual directa, que utiliza gráficos divertidos e instrucciones fáciles de seguir, para aprender a construir proyectos en Scratch.

El autor Jon Woodcock estudió Física en la Universidad de Oxford y cuenta con un doctorado en Astrofísica Computacional de la Universidad de Londres. Ha realizado simulaciones espaciales y robots inteligentes, así como investigaciones en compañías de alta tecnología. Aprendió a programar desde los 8 años, y ha programado desde microcontroladores hasta supercomputadoras. Tiene pasión por la ciencia y la educación tecnológica, es autor de varios libros de esos tópicos y le gusta crear clubes de programación en las escuelas. Con el sello editorial de Dorling Kindersley (DK) tiene varios libros de programación en Scratch y de fundamentos de programación dirigidos a niños en educación básica.

El libro consta de 11 capítulos y un pequeño glosario con palabras técnicas. De los 11 capítulos, 8 están enfocados a realizar un juego completo en cada uno, donde se van presentando de forma práctica los diversos conceptos de programación y las características del lenguaje Scratch y su entorno.

El capítulo 1 aborda en forma general los juegos de computadora. Primero menciona los elementos que hacen a un juego bueno, estos son: los personajes, los objetos dentro del juego, la mecánica del juego, las reglas, los objetivos, los controles, la dificultad, el mundo donde se desarrolla el juego y enfatiza el concepto de jugabilidad. Después menciona que otro de los elementos importantes es la atmósfera del juego, la cual está formada por la historia del juego, los sonidos, la velocidad o ritmo del juego, los gráficos, la paleta de colores y los escenarios. El capítulo continúa con los tipos o géneros de juegos y finaliza con una introducción de cómo funciona la programación, dándole instrucciones a la computadora en un lenguaje que ella entienda, esto es, a través de los lenguajes de programación.

El capítulo 2 da una introducción al vocabulario utilizado en el lenguaje Scratch; por ejemplo, a los objetos que interactúan en el juego se les llama *sprites* y al conjunto de bloques de instrucciones que le indican al *sprite* qué hacer se les llama *scripts*. También se explica cómo instalar Scratch para un uso sin internet o cómo utilizarlo desde el sitio oficial <https://scratch.mit.edu/> a través del registro en línea. La versión en línea de Scratch puede utilizarse en Windows, Mac y Ubuntu; la versión fuera de línea solo soporta

los dos primeros; por el momento solo se puede utilizar en computadoras personales y portátiles. Por último, se presenta el entorno de desarrollo de Scratch. Esta edición en particular del libro fue realizada con la versión 2.0 de Scratch. La edición 2019 del libro se basa en la versión 3.0.

Es importante mencionar que para esta reseña se realizaron los 8 proyectos del libro utilizando la versión actual de Scratch (3.0), pero las instrucciones del libro son para la versión 2.0. El entorno de Scratch en la versión 3.0 sí cambió con respecto al de la 2.0 y algunas opciones fueron omitidas o reacomodadas. El mayor cambio es en el entorno de dibujo de los *sprites*, y al principio cuesta trabajo seguir las instrucciones del libro por dichos cambios. El otro problema detectado es el cambio de los *sprites* y fondos provistos en el lenguaje; los sonidos no cambiaron. Los gráficos de los *sprites* en 2.0 son más caricaturescos y pixeleados, los de 3.0 son más reales y mejor definidos, aunque no necesariamente mejores o más agradables para los niños. Especialmente en el juego 5 se perdía la esencia del juego al utilizar los nuevos *sprites*, así que se hizo la prueba de instalar la versión 2.0, copiar a disco los *sprites* y fondos necesarios, y cargarlos en la versión 3.0. La prueba fue exitosa y también se reporta que ambas versiones conviven sin problema en una misma computadora.

El objetivo de los capítulos 3 al 10 es realizar un proyecto diferente de un juego completo y mediante la práctica ir aprendiendo a la par conceptos de programación. Todos estos capítulos están estructurados igual: primero se explica la mecánica del juego, su objetivo, personajes, escenario y controles; después presenta paso a paso cómo realizar el juego, mostrando absolutamente todos los *scripts* necesarios, indicando el porcentaje de progreso hasta que se concluye el 100% del juego; una vez terminado el juego se dan consejos para mejorarlo, algunos con *scripts*. Las lecciones utilizan ejemplos visuales, instrucciones fáciles de seguir (aunque dependientes de la versión de Scratch) e incluye arte visual en forma de personajes animados que puede resultar divertido para los lectores más pequeños.

A lo largo de la explicación se incluyen varias cápsulas informativas de distintos tipos para una enseñanza más completa; hay cápsulas de jerga, que

presentan palabras técnicas de programación (*bugs*, expresiones booleanas, eventos, funciones, banderas y variables locales y globales); cápsulas de consejos de expertos, que explican cómo utilizar ciertos bloques de instrucciones o cómo resolver ciertos problemas (manejo de diversos ciclos y condicionales, generación de números aleatorios, uso de coordenadas y grados, detección de colisiones, redimensionamiento de imágenes, creación de clones u objetos, operadores lógicos y aritméticos, simulación de la gravedad, manejo de variables, creación de respaldos, juegos por ciclos, solicitud de datos al usuario, manejo de listas, envío de mensajes y depuración); cápsulas de diseño de juegos, que explican conceptos propios de los videojuegos (sus tipos de controles o mandos, música, espaciado, historia, física, animación, temática, tipos de saltos, tipos de detección de colisiones, diseño de niveles y ángulos de la cámara).

El juego 1 en el capítulo 3 se llama “Star Hunter”. Consiste en un personaje que debe capturar estrellas mientras esquiva pulpos en un mundo submarino. En este juego se aprende a utilizar Scratch, a controlar un personaje con el ratón, poner imágenes de fondo y sonidos para el escenario, varios movimientos autónomos y aleatorios, detección simple de colisiones y manejo de puntajes, por mencionar algunas.

El juego 2 en el capítulo 4 se llama “Cheese Chase”. Consiste en un ratón en un laberinto que debe capturar su queso, mientras evita tocar a escarabajos que corren en el laberinto y a un fantasma que lo persigue. En este juego se aprende a controlar un personaje con las teclas de dirección, a usar el editor de dibujo de Scratch, a hacer laberintos, enviar mensajes, manejo del mejor puntaje y a colocar pantallas de fin de juego, por mencionar algunas.

El juego 3 en el capítulo 5 se llama “Circle Wars”. Consiste en que el jugador es un círculo azul y captura círculos verdes mientras evita círculos rojos que lo persiguen. En este juego se aprende a crear clones (instanciar objetos) y hacer *scripts* para controlarlos o desaparecerlos, manejar el escenario de victoria y derrota, agregar cronómetro al juego y poner pantallas de instrucciones del juego, por mencionar algunas.

El juego 4 en el capítulo 6 se llama “Jumpy Monkey”. Consiste en un mono que es lanzado por

los aires para capturar racimos de plátanos, controlando mediante teclado la velocidad y el ángulo de lanzamiento. En este juego se aprende a programar eventos de teclado y simular movimiento con efecto de la gravedad, principalmente.

El juego 5 en el capítulo 7 se llama “Doom on the Broom”. El juego consiste en una bruja con un movimiento rotario que lanza bolas de fuego a diversos enemigos que intentan atraparla. En este juego se aprende a hacer animaciones de personajes, movimientos con ángulos, manejo de vidas del jugador, transparencias y cambios de colores, así como aumento de dificultad; todo mientras se repasa el manejo de clones y movimientos autónomos.

El juego 6 en el capítulo 8 se llama “Dog’s Dinner”. Es un juego de plataformas donde un perro recorre varios niveles con obstáculos y recompensas. En este juego se aprende a hacer bloques propios de instrucciones (funciones), a desplazarse con plataformas, a saltar, a detectar colisiones mediante un *sprite* oculto, a manejar niveles y a tener un control maestro del juego; todo mientras se repasa la creación de dibujos propios, manejo de variables, sonidos y animaciones. Es el juego más completo y difícil del libro (y el favorito del autor, según una entrevista publicada en <https://prbythebook.com/jon-woodcock/>).

El juego 7 en el capítulo 9 se llama “Glacier Race”. El juego consiste en una carrera de autos para dos jugadores, a través de un escenario de hielo con desplazamiento (*scrolling*) vertical. En este juego se aprende a controlar el juego completo mediante un ciclo y recepción de mensajes, a hacer escenarios con desplazamiento, a tener un personaje para dar instrucciones, solicitar información al usuario y decir el resultado de la carrera, por mencionar algunas.

El juego 8 en el capítulo 10 se llama “Tropical Tunes”. El juego consiste en cuatro tambores que generan una secuencia musical al azar, que el jugador debe imitar para seguir en el juego, y en casa ronda la secuencia va aumentando. En este juego se aprende a utilizar mejor las variables locales, a generar notas musicales, a manejar listas de datos y a tener un control maestro para el juego que genera eventos autónomos, por mencionar algunas.

El capítulo 11 promueve la creación de proyectos propios, modificando los que ya existen en la plataforma en línea de Scratch. Presenta las herramientas de dicha plataforma para copiar proyectos o *sprites*, y compartir proyectos propios. Después, se presentan consejos para mejorar la programación, como poner comentarios, mejorar la nomenclatura de los *sprites*, variables y mensajes, y no utilizar números fijos, sino variables. También, presenta las herramientas de ayuda de Scratch. Continúa el capítulo con consejos para mejorar el diseño y programación de juegos, dentro y más allá de Scratch, así como actividades para fomentar la creación de juegos. Se explican los trabajos que existen en la industria de los videojuegos, incluyendo escritores, diseñadores de juegos, productores, artistas, compositores, diseñadores de sonidos, programadores y probadores; y las versiones por las que pasa un juego antes de liberarse. Para alentar el trabajo independiente y de un solo programador, se menciona el caso del juego Minecraft, creado por el sueco Markus Persson en 2009, que en 5 años se volvió uno de los juegos más populares con más de 100 millones de usuarios registrados y esto provocó su venta a Microsoft por 2,500 millones de dólares.

En conclusión, es un excelente libro para aprender a programar en Scratch desde cero y también sirve para aprender lo básico de videojuegos. Respeta la filosofía de Scratch de Programar para aprender, ya que es totalmente práctico, no explica temas antes de aprovecharlos, no contiene texto extra, todo lo escrito durante el paso a paso son instrucciones que necesita el usuario y los proyectos son lo suficientemente variados como para aprender la mayoría de las instrucciones de Scratch, pero no todas, como la edición de sonidos o el manejo de la cámara web. El mismo autor hizo otros libros de trabajo para aprender más sobre Scratch y los proyectos no se repiten. Se recomienda que los proyectos se desarrollen en el orden que se presentan, ya que hay ciertos conceptos e instrucciones que se usan en más de un proyecto, pero solo la primera vez profundiza en la explicación. Algo que debe ser reconocido de este libro es que no omite un solo script para llegar al 100% de avance del juego, aunque el script sea similar a uno ya visto en otro proyecto previo o incluso dentro del mismo proyecto. Como ya se mencionó, el único inconveniente para aprender a utilizar Scratch utilizando este libro

fue el cambio de versiones del lenguaje, y lo más trabajoso fue el uso del editor de imágenes incluido en Scratch, ya que no tiene muchas herramientas como un Paint, por ejemplo, y le quitaron características en la versión 3.0; la recomendación sería realizar los dibujos con otro software y luego importarlos a Scratch, es más rápido. Este libro puede servir como introducción para que los niños y jóvenes aprendan a codificar, a pensar creativamente y razonar sistemáticamente, las cuales son habilidades cada vez más necesarias ahora que la educación STEM (Ciencia, Tecnología, Ingeniería y Matemáticas) está tomando mayor importancia en el mundo.

Para finalizar, quiero agradecer al Programa para el Desarrollo Profesional Docente, para el Tipo Superior (PRODEP), ya que este libro fue adquirido gracias al apoyo económico otorgado en el reconocimiento al perfil deseable de los profesores de tiempo completo.

Autor de la reseña:

Manuel Alejandro Valdés Marrero

Programador Certificado en Java
Universidad del Mar campus Puerto Escondido
valdes@zicatela.umar.mx

Recibido: 22 de octubre de 2019

Aceptado: 10 de julio de 2020