

CURSO DE PROMOCIÓN Y EXTENSIÓN EDUCATIVA

Conceptos básicos de programación informática y pensamiento computacional con *Scratch*

(60 horas)

1. JUSTIFICACIÓN

Los avances tecnológicos que han tenido lugar en los últimos años han revolucionado la sociedad, que ha tenido que adaptarse y acomodarse a ese progreso aprendiendo a manejar unas herramientas tecnológicas que hasta hace poco se desconocían.

El mismo hecho de estar ahora mismo leyendo este texto ha sido posible gracias a que alguien, en algún momento, aprendió a programar. En la actualidad, estamos rodeados de dispositivos electrónicos que utilizamos para todo tipo de tareas. En algunos casos, resulta más obvio que alguien ideó un programa, por ejemplo, cuando enviamos un mensaje en WhatsApp. En otros casos, no nos paramos a pensar que hay un programa detrás del aparato que tenemos ante nosotros. Por ejemplo, para que nuestro microondas funcione, también alguien programó una secuencia de instrucciones. O para que vuele un avión. O para hacer una transferencia. O para hacer funcionar el contador de la electricidad, o los botones del ascensor. Todo esto es posible gracias a la programación informática.

Sin embargo, la finalidad de este curso no es aprender a programar como tal, sino comprender qué es programar y cómo el hecho de tener ciertos conocimientos de programación puede servir para fomentar la lógica y la resolución de problemas de la vida del día a día. El pensamiento computacional es un elemento transversal a las competencias matemática, científica y tecnológica, que encuentra también su aplicación a otros campos del saber como la lógica y la filosofía. Cada vez son más los entornos tecnológicos que facilitan plasmar el pensamiento computacional en un programa, sin necesidad de poseer conocimientos avanzados de programación.

Scratch, creado por el Instituto Tecnológico de Massachusetts, permite programar utilizando bloques visuales para resolver ciertos problemas, crear animaciones, diseñar un applet web, dotar de funcionalidad a un robot o, incluso, diseñar objetos tridimensionales que podrán ser impresos.

El pensamiento computacional se enmarca perfectamente en lo que se denomina *cultura maker*, a veces también conocida como *movimiento maker* o, incluso, como la *tercera revolución industrial*, y está relacionada también con el diseño o modelado y la impresión en 3D. Esta cultura representa una extensión del concepto DIY (Do it Yourself o hágalo-usted-mismo), promoviendo la idea que todo el mundo es capaz de desarrollar cualquier tarea en vez de contratar a un especialista para realizarla. De esta forma desarrolla la competencia de aprender a aprender.

Es una cultura inspiradora y que cultiva a la vez los valores de la cultura libre, constituyendo una comunidad en la que todo el mundo se inspira y beneficia de las contribuciones de los demás.

El alumnado del curso precisa de antemano de conocimientos y competencias necesarias básicas sobre manejo del ordenador como navegar por Internet, buscar información, descargar y subir archivos, abrir y cerrar programas, guardar archivos, enviar correos electrónicos con archivos adjuntos, añadir mensajes en foros, etcétera.

2. RELACIÓN CON LAS COMPETENCIAS CLAVE

Con este curso, el alumnado desarrolla gran parte de las competencias clave:

La competencia en comunicación lingüística se desarrolla utilizando el lenguaje específico. El lenguaje, en sentido amplio, se refiere a todos aquellos medios de expresión, gráficos, simbólicos, gestuales, etc., con los que transmitimos y recibimos información. Los bloques visuales de programación, así como sus instrucciones básicas, constituyen una forma de lenguaje que sirve como soporte al pensamiento computacional.

El pensamiento computacional requerido permite abordar situaciones desde el punto de vista algorítmico. La competencia matemática y las competencias básicas en ciencia, tecnología e ingeniería inducen y fortalecen algunos aspectos esenciales de la formación de las personas que resultan fundamentales para la vida. La consecución y sostenibilidad del bienestar social exige conductas y toma de decisiones personales estrechamente vinculadas a la capacidad crítica y visión razonada y razonable de las personas.

La competencia personal, social y de aprender a aprender es muy relevante, requiere tomar conciencia de las propias capacidades y cómo desarrollarlas, siendo capaz de autoevaluarse, plantearse metas alcanzables y trabajar de forma cooperativa. Aprender a aprender es una de las señas de identidad de la *cultura maker*, uno de los hábitats naturales del pensamiento computacional hoy en día.

La competencia emprendedora implica la capacidad de transformar las ideas en actos. Significa adquirir conciencia de la situación a intervenir o resolver, y saber elegir, planificar y gestionar los conocimientos, destrezas y actitudes con criterio propio, con el fin de alcanzar el objetivo previsto. Las oportunidades que ofrece la *cultura maker* son diversas, desde hacer realidad proyectos personales a desarrollar una carrera profesional.

Los contenidos del curso permiten trabajar de forma directa la competencia digital, pues la *cultura maker*, en general, y el pensamiento computacional en particular se enfocan hacia la resolución creativa de problemas de forma colaborativa. El aprendizaje a través del hacer y la colaboración entre compañeros son dos de sus características esenciales, en un entorno actual y dinámico que resulta esencial conocer.

La competencia ciudadana al trabajar con una herramienta libre, *Scratch*. En el ámbito de la ingeniería informática, *scratching* significa reutilizar código, el cual puede ser utilizado de forma beneficiosa y efectiva para otros propósitos y fácilmente combinado, compartido y adaptado a nuevos escenarios. Cualquier usuario puede descargarse y trabajar sobre proyectos públicos subidos y desarrollados por otros usuarios. El conocimiento es libre.

La competencia en conciencia y expresiones culturales incorpora un componente expresivo referido a la propia capacidad estética y creadora y al dominio de aquellas capacidades relacionadas con los diferentes códigos artísticos y culturales, para poder utilizarlas como medio de comunicación y expresión personal.

3. ASPECTOS INNOVADORES

El curso pretende paliar la discrepancia entre la oferta de formación existente sobre nuevas tecnologías (robótica, pensamiento computacional, impresión 3D, etc.) para el alumnado menor de 18 años y la ausencia de opciones para personas adultas que, sin pretender realizar formación especializada, sienten curiosidad por conocer qué supone todo este nuevo conjunto de tecnologías.

También se pretende desterrar la idea de que la programación es un saber reservado a unos pocos, que requiere de grandes conocimientos o que únicamente soluciona problemas puramente informáticos.

Para fomentar la participación en el *movimiento maker* no se necesitan mayores conocimientos técnicos, sino ganas de aprender a aprender.

4. OBJETIVOS

Los objetivos generales que se plantean con este curso son:

- Distinguir entre pensamiento computacional y programación.
- Conocer los fundamentos del pensamiento computacional.
- Conocer rudimentos básicos comunes a todos los lenguajes de programación.
- Utilizar el pensamiento computacional para la resolución de problemas.
- Comprender y practicar conceptos básicos de programación y construir programas sencillos.
- Favorecer indirectamente la empleabilidad de las personas con conocimientos básicos del uso de las tecnologías y el ordenador.

5. CONTENIDOS

Los contenidos a desarrollar en el curso son los siguientes:

- Fundamentos del pensamiento computacional y resolución de problemas simples.
- Lenguajes de programación. ¿Qué es programar?
- Algoritmos y diagramas de flujo.
- Programación mediante Scratch. Bloques. Objetos, disfraces y fondos. Programas y movimiento. Lápiz y sonido. Control: condiciones y repeticiones. Mensajes y variables.
- Proyecto y primer juego.
- Cultura libre. Reutilización de código. Licencias.
- Tecnologías web aplicadas al pensamiento computacional.

6. METODOLOGÍA

Las actividades tendrán como punto de partida los intereses del alumnado, sus experiencias y conocimientos previos, de forma que se favorezca el aprendizaje significativo, siendo el alumnado el protagonista del aprendizaje. Se potenciará la autoestima con el respeto a los ritmos de aprendizaje, resolviendo entre todos -con la guía del docente- los problemas que se vayan presentando.

Se partirá de la presentación del pensamiento computacional, a partir de la idea de pseudocódigo, identificando cómo realmente está presente en la vida cotidiana: al hacer una receta de cocina, un entrenamiento deportivo, etc. Se traducirá este pseudocódigo a algunos lenguajes de programación sencillo, para ver sus similitudes. Seguidamente se introducirá el entorno de Scratch a partir de ejemplos sencillos. Finalmente, se dedicará gran parte a realizar programas según los intereses del grupo y donde intervengan todos los tipos de bloques lógicos.

Cuando se trate de reutilizar el código de otro usuario o de un compañero del curso, se incidirá en reforzar el espíritu crítico, reflexionando sobre la fiabilidad de la información disponible, los valores éticos que deberían regir nuestras vidas para hacer un uso de las TIC más responsable y enriquecedor. En definitiva, participar del *movimiento maker* y del conocimiento libre.

Se fomentarán los trabajos colaborativos al trabajar con herramientas que lo permiten. Será de relevante importancia el fomento de aprendizajes significativos que sean de utilidad para la empleabilidad y el emprendimiento.

7. DURACIÓN DEL CURSO

El curso tendrá una duración total de 60 horas, distribuidas según las necesidades del alumnado y la disponibilidad horaria del centro, de forma bimestral, cuatrimestral o anual.

8. MATERIALES

- Aula de informática.
- Ordenadores con sistema operativo de entorno gráfico.
- <https://catedu.github.io/ensena-pensamiento-computacional-con-scratch/>
- <http://formacion.educalab.es/course/view.php?id=529>

9. EVALUACIÓN

9.1. CRITERIOS DE EVALUACIÓN Y SU CONCRECIÓN

- Conocer los conceptos y términos básicos de la programación informática.
- Conocer las diversas opciones de Scratch.
- Programar aplicaciones sencillas con Scratch.
- Utilizar y reutilizar con criterio programas ya hechos por la comunidad.
- Conocer y utilizar las licencias de creación y uso de contenidos (materiales propios, *Creative Commons*, *Copyleft* y *Copyright*).

Para precisar estos criterios de evaluación, se verificará el nivel de adquisición de los contenidos en base a estos indicadores de concreción:

Adquisición insuficiente	No ha sido alcanzado un mínimo aceptable y necesita una mejora sustancial.
Adquisición básica	Ha sido alcanzado un mínimo aceptable, aunque es susceptible de mejora.
Adquisición excelente	Se evidencia una adquisición excepcional, por encima de los mínimos.

9.2. PROCEDIMIENTOS E INSTRUMENTOS DE EVALUACIÓN

La evaluación de las producciones del alumnado por parte del docente permite adecuar los retos planteados en cada momento y proponer una secuencia de aprendizaje que tenga en cuenta la capacidad y los intereses del grupo de clase.

Los procedimientos e instrumentos de evaluación serán variados, teniendo en cuenta la actitud activa ante el aprendizaje, así como el trabajo llevado a cabo en el aula y una verificación de conocimientos, poniendo el acento en la superación personal y evitando en la medida de lo posible la competitividad y la valoración excesiva de las calificaciones.

9.3. CRITERIOS DE CALIFICACIÓN

Sobre los instrumentos antes referidos, se valorará, en primer lugar, que la asistencia a clase supere el 50% para que el alumnado pueda ser evaluado, de acuerdo con los siguientes porcentajes:

- Conocimientos adquiridos: 40%
- Trabajo realizado: 40%
- Actitud activa y participativa: 20%