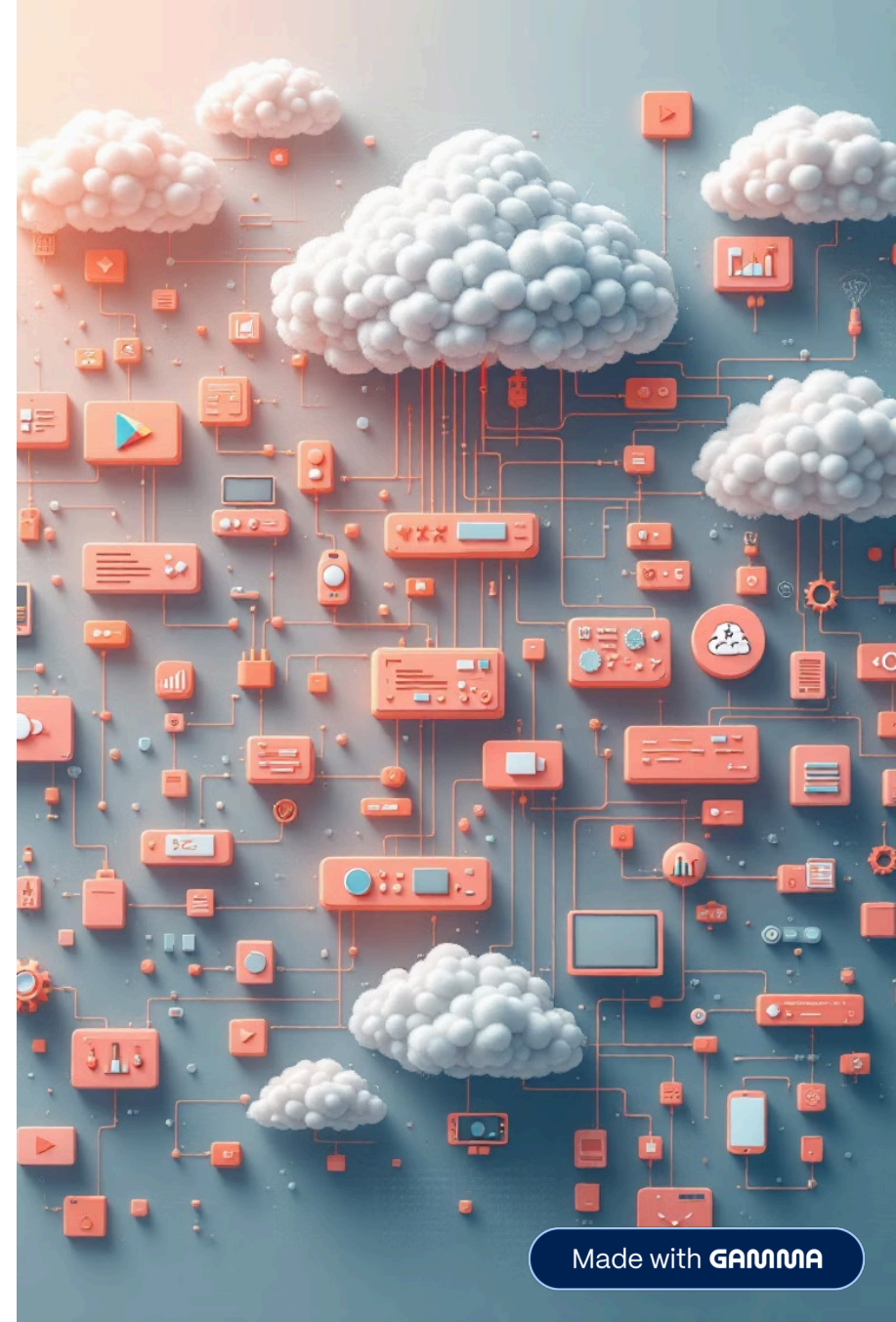


Tema: Pensamiento computacional, programación básica y IoT

Convertir ideas en programas y conectar dispositivos con sensores suena a magia, pero es pura lógica y creatividad. En este capítulo te adentrarás en el **pensamiento computacional**, aprenderás a escribir **algoritmos** con diagramas de flujo y pseudocódigo, crearás proyectos con **programación por bloques** en Scratch y llevarás tus programas al mundo físico con **BBC micro:bit** y los primeros pasos en **IoT** (Internet de las Cosas).



1. Pensamiento computacional: descomponer y abstraer

Cuando te enfrentas a un reto —por ejemplo, programar un juego de laberinto en Scratch—, primero debes:

Descomposición

Dividir el problema en partes más pequeñas: movimiento del personaje, detección de paredes, conteo de pasos y condición de llegada.

Abstracción

Identificar qué información es relevante (posiciones en la cuadrícula, dirección de movimiento) y descartar lo accesorio (color de fondo, estilo de sprite).

Al dominar estas habilidades, cualquier algoritmo que escribas será más claro y fácil de mantener.

2. Algoritmos y pseudocódigo: planificando tu programa

Antes de tocar bloques o código, traduce tu lógica a un **diagrama de flujo** o **pseudocódigo**:

Diagramas de flujo

Usa símbolos estándar: óvalo para inicio/fin, rectángulo para procesos, rombo para decisiones y flechas para el flujo de control.

Pseudocódigo

Escribe pasos en lenguaje cercano al natural:

```
INICIO
  repetido hasta que X llegue a meta
    mover X según tecla pulsada
    si X toca pared
      deshacer movimiento
    fin si
  fin repetido
FIN
```

Esta planificación previene errores y acelera la posterior codificación.

3. Programación por bloques con Scratch

En Scratch, arrastras y encajas piezas de código:

01

Sprites y escenarios

Elige o dibuja personajes y fondos.

02

Eventos de inicio

Bloques como "Al hacer clic en bandera verde" disparan tu programa.

03

Control de flujo

Bucles repetir X veces, por siempre y condicionales si... entonces.

04

Variables

Crea contadores para puntuaciones o pasos.

05

Mensajes y sonidos

Usa enviar mensaje para comunicar sprites y añadir efectos.

4. Introducción a MakeCode y BBC micro:bit

La BBC micro:bit es una placa con LEDs, botones y sensores. En MakeCode:



Entradas

Lee botones (si botón A presionado), sensores de luz o temperatura.



Salidas

Muestra un patrón en la matriz de LEDs o envía un mensaje por radio.



Variables y funciones

Almacena valores de sensor y reutiliza bloques.



Depuración

Prueba en el simulador antes de cargar en la placa.

5. Primeros pasos en IoT: enviar datos a la nube

El Internet de las Cosas te permite:



Conectar micro:bit a Wi-Fi o radio

Usa extensiones de MakeCode como WiFi o Radio.



Configurar un servicio cloud

Crea una cuenta gratuita en plataformas como ThingSpeak o Adafruit IO.



Enviar datos periódicos

Dentro de un bucle, publica temperatura o nivel de luz a un canal.



Visualizar

La plataforma genera gráficos en tiempo real y te envía alertas si los valores superan umbrales.

6. Depuración y mejora continua

La **depuración** es parte esencial del aprendizaje:

Pruebas unitarias

Verifica bloques o funciones por separado.

Mensajes de log

Muestra valores intermedios en consola o LEDs.

Iteración

Corrige un error, prueba de nuevo y comprueba el cambio.

Registro de errores

Anota cada fallo y solución para futuras referencias.

Con paciencia y reflexión, cada bug superado refuerza tu confianza y tus habilidades.

¡Empieza tu proyecto ahora!

Con este viaje que va del pensamiento lógico a la interacción con el mundo físico y la nube, estarás listo para diseñar, programar y conectar dispositivos inteligentes, dominando las bases de la programación y los principios del IoT. ¡Empieza tu proyecto ahora!

