

## **Mi ciudad lo tiene**

Grado sugerido: Octavo

**Juan Manuel García Suarez**

*Publicado en el Banco Virtual de Recursos de Colombia Programa en el año 2025.*

Este material se comparte bajo la licencia Creative Commons Atribución-NoComercial-SinDerivadas 4.0 Internacional (CC BY-NC-ND 4.0). Puede copiar y redistribuir el material en cualquier medio o formato, siempre que dé el crédito adecuado al autor, no lo use con fines comerciales, y no remezcle, transforme o cree a partir del material.

Para más información, consulte la licencia completa en [Deed - Attribution-NonCommercial-NoDerivatives 4.0 International - Creative Commons](#)

Para contactar al autor/a de este recurso, escriba a: [juangarcia@ensn.ie.edu.co](mailto:juangarcia@ensn.ie.edu.co)

## PLANTILLA SECUENCIA DIDÁCTICA

Este documento presenta una planeación de una sesión de clases que incorpore algún tipo de actividad para el desarrollo del pensamiento computacional. Se estima que el desarrollo de la actividad propuesta en este documento no supere los 120 minutos.

Tenga en cuenta que la plataforma solo recibirá recursos en formato **.pdf** cuyo tamaño no exceda los **10MB de peso y las 20 páginas de extensión**.

<b>Aprendizaje(s) esperado(s)</b>	<i>Indique el o los aprendizajes que busca desarrollar en las/los estudiantes durante la sesión de clase</i>
	<ul style="list-style-type: none"><li>• Reconocer elementos representativos de su ciudad y seleccionar los más significativos para un juego visual.</li><li>• Aplicar principios del pensamiento computacional como descomposición, patrones y abstracción en el diseño del juego.</li><li>• Utilizar herramientas de inteligencia artificial para generar imágenes personalizadas de forma ética y creativa.</li><li>• Comprender el funcionamiento básico de un algoritmo para construir las cartas del juego <i>Spot It</i>.</li><li>• Crear un conjunto funcional de cartas tipo <i>Spot It</i> en formato físico o digital.</li><li>• Participar activamente en la producción y prueba del juego, fortaleciendo habilidades colaborativas y de comunicación.</li><li>• Valorar su identidad local y expresar sentido de pertenencia a través de una creación lúdica y significativa.</li></ul>
<b>Materiales requeridos</b>	<p><i>Liste todos los materiales que se requieren para completar las actividades propuestas para la sesión de clase</i></p> <ul style="list-style-type: none"><li>• Computadores con acceso a internet y sistema operativo Windows 11</li><li>• Acceso a herramientas de generación de imágenes por IA (ej. Bing Creator, Craiyon, DALL·E)</li><li>• Entorno de programación visual o en línea (por ejemplo, Scratch, Google Colab, repl.it, etc.)</li><li>• Aplicación o código para generar las cartas del juego (<i>Spot It generator</i> o algoritmo adaptado en Python)</li><li>• Papel tipo cartulina o papel adhesivo para impresión</li><li>• Tijeras, regla, cinta o pegamento</li><li>• Marcadores, colores y sobres plásticos (para organización de cartas)</li><li>• TV o proyector para orientación grupal</li></ul>

<b>Conocimientos previos requeridos</b>	<p><i>Indique los conocimientos y habilidades que deberían tener de forma previa sus estudiantes con el fin de desarrollar exitosamente las actividades que propone</i></p> <ul style="list-style-type: none"> <li>• Comprensión básica del uso del navegador web y búsqueda segura de información</li> <li>• Conocimiento elemental del manejo de carpetas y archivos en el computador</li> <li>• Nociones de dibujo o representación gráfica (preferiblemente con actividades previas en artística)</li> <li>• Familiaridad con juegos de mesa o dinámicas visuales como <i>Dobble</i>, <i>Spot It</i> u otros similares</li> <li>• Idealmente, haber participado en experiencias introductorias al pensamiento computacional o actividades desconectadas como reconocimiento de patrones, emparejamiento o clasificación de elementos</li> </ul>
<b>Actividad(es) a desarrollar</b>  <i>Indique las acciones que realizarán el/la docente y sus estudiantes y las indicaciones si el trabajo se debe realizar de forma individual, en parejas o grupal.</i>	<b>Tiempo estimado</b>  <i>Minutos o porcentaje</i>
1. INICIO: breve presentación sobre los juegos visuales tipo <i>spot it</i> y su lógica matemática. se ejemplifica con 2 cartas (modelo n=6). reflexión guiada sobre qué representaciones visuales hacen parte de la identidad de <i>su ciudad</i> .	20 minutos
2. DESARROLLO: en parejas, los estudiantes listan ítems representativos de su ciudad: monumentos, comidas típicas, celebridades, expresiones culturales, flora y fauna local, etc. Posteriormente, exploran herramientas de generación de imágenes con IA (como bing creator o canva ia), con acompañamiento docente. Cada pareja selecciona al menos 6 ítems para ilustrar con ia.	40 minutos
3. APLICACIÓN: el docente les comparte un entorno básico de programación (scratch, pseint o google colab con python) ya configurado para generar cartas. Los estudiantes, con ayuda, insertan sus ítems y observan cómo el programa genera una plantilla imprimible. Se reflexiona brevemente sobre cómo funciona el algoritmo.	40 minutos
4. CIERRE y TAREA: Cada grupo descarga su plantilla. Como tarea, deben imprimir y recortar las cartas para llevarlas a clase en la siguiente sesión. El juego será utilizado como cierre del proceso. Se proyectan algunas cartas generadas como ejemplo.	20 minutos
<b>Adaptaciones</b>	
<i>Acá se brindan las sugerencias o recomendaciones para adaptaciones a diversos contextos (ejemplo: zona rural, población con discapacidad o sin acceso a Internet)</i>	

### **Contextos sin conectividad a Internet**

- El docente puede llevar impresas una selección de imágenes representativas de la ciudad (previamente recopiladas y organizadas en una carpeta).
- En lugar de usar IA para generar imágenes, los estudiantes pueden dibujarlas o recortarlas de revistas o periódicos.
- Si no se cuenta con acceso a entornos de programación, el docente puede entregar una plantilla impresa en la que los estudiantes dibujen directamente los ítems siguiendo la lógica del juego.

### **Zonas rurales con recursos limitados**

- Se sugiere trabajar por grupos y usar un solo equipo con recursos mínimos para proyectar ejemplos.
- Las cartas pueden diseñarse a mano y recortarse con materiales reutilizables (cartulina reciclada, hojas usadas, etc.).
- El enfoque se adapta al fortalecimiento del pensamiento lógico y a la creatividad en el diseño manual.

### **Población con discapacidad auditiva**

- La explicación de las instrucciones se puede acompañar con lenguaje de señas y/o material visual claro (infografías paso a paso).
- En el momento de generar ítems representativos de la ciudad, se promueve el uso de imágenes y símbolos fácilmente interpretables para facilitar la comunicación visual.
- En actividades grupales, se sugiere el uso de pizarras o papeles grandes para facilitar la expresión colaborativa y visual de ideas.

### **Población con discapacidad visual parcial o total**

- Para estudiantes con baja visión, se puede usar software con alto contraste y tipografías aumentadas.
- Si hay estudiantes con ceguera total, se puede adaptar la dinámica utilizando tarjetas en relieve o con texturas diferentes para cada ítem. También se puede transformar la actividad en un juego auditivo con pistas sonoras representativas de los elementos culturales.

### **Actividades evaluativas**

*Describe la forma en que un(a) docente que siga esta secuencia didáctica podría evaluar que sus estudiantes estén alcanzando los aprendizajes propuestos para la sesión de clase*

1. **Rúbrica de proceso de diseño:** El docente podrá utilizar una rúbrica sencilla para valorar aspectos como:
  - Comprensión del propósito del juego y del concepto de ítems representativos.
  - Participación activa en el análisis y selección de elementos culturales de la ciudad.
  - Creatividad y pertinencia en las imágenes generadas (IA o dibujo).
  - Exactitud en la relación entre cartas e ítems dentro del modelo Spot It (lógica combinatoria).
2. **Revisión del prototipo de cartas:**

<ul style="list-style-type: none"> <li>○ Se evalúa que cada carta contenga exactamente 6 ítems y que cada par de cartas comparta un único ítem común, conforme a la lógica del modelo de juego.</li> <li>○ Se valora el diseño visual, la organización y el esfuerzo por presentar un conjunto coherente.</li> </ul> <p><b>3. Juego final en grupos:</b></p> <ul style="list-style-type: none"> <li>○ Durante la fase de juego, el docente observará la participación de los estudiantes, la comprensión de reglas y el uso del pensamiento lógico para identificar coincidencias entre cartas.</li> <li>○ Se podrá utilizar una lista de cotejo para anotar aspectos como colaboración, agilidad mental, respeto por turnos y uso del vocabulario aprendido.</li> </ul> <p><b>4. Autoevaluación y metacognición:</b></p> <ul style="list-style-type: none"> <li>○ Se sugiere aplicar una breve encuesta escrita o verbal donde los estudiantes reflexionen sobre lo aprendido, los desafíos que enfrentaron y qué mejorarían en una futura edición del juego.</li> </ul>	
<b>Referencias</b>	<i>Liste los recursos consultados para la creación de este recurso. Preferiblemente siga el formato APA7.</i>

## ANEXO

*Incluya los anexos requeridos aquí. Si son videos, presentaciones u otros materiales, ingrese un enlace y/o un código QR que permita accederlos libremente.*

### ANEXO 1. Pseudocódigo

Para generar las cartas se utiliza el siguiente pseudocódigo:

#### INICIO

1. DEFINIR  $N = 5$
2. CALCULAR número de cartas =  $N^2 + N + 1 = 31$
3. CARGAR imágenes desde carpeta "imagenes/" → ListaDeImágenes[1...31]
4. GENERAR estructura de cartas (matemáticamente):
  - CADA carta debe tener 6 ítems
  - TODAS las combinaciones posibles entre cartas deben compartir solo UN ítem

USAR algoritmo estándar para construcción de cartas Spot It con  $n=6$

5. POR CADA carta EN estructura\_de\_cartas:
  - a. CREAR una nueva imagen en blanco o plantilla con fondo
  - b. UBICAR las 6 imágenes correspondientes (usando una biblioteca de edición de imágenes como Pillow (Python))
    - Ajustar tamaño (por ejemplo, 100x100 píxeles)
    - Ubicar en círculo o patrón visual claro

- c. GUARDAR la imagen de la carta en carpeta "salida/"
6. GENERAR un PDF con las 31 cartas para imprimir
- UNA carta por página o 4 por hoja

FIN

Este pseudocódigo permite a los estudiantes comprender el flujo lógico del programa, independientemente del lenguaje de programación, facilitando el desarrollo de habilidades de abstracción y diseño algorítmico.

## ANEXO 2. Código Python

El siguiente enlace tiene el código fuente completo en Python para su consulta y ejecución con un total de 31 cartas, 6 elementos por cartas y  $n=5$  ( $n$  es el orden del plano proyectivo finito)

[https://drive.google.com/file/d/1rcSFr\\_W0pEfdkICWqPMWLrQ8remcxSPp/view?usp=sharing](https://drive.google.com/file/d/1rcSFr_W0pEfdkICWqPMWLrQ8remcxSPp/view?usp=sharing)

### Requisitos técnicos

#### Carpetas necesarias:

dobble\_generator.py

/

```
|— imagenes/  
|   |— simbolo_01.png  
|   |— simbolo_02.png  
|   ... hasta 31 imágenes  
|— cartas/ (se creará automáticamente)
```

### Instalación de dependencias:

pip install pillow reportlab

### Salida del programa

- Imágenes guardadas en **./cartas/**: **carta\_01.png**, ..., **carta\_31.png**
- PDF generado: **dobble.pdf** con 4 cartas por página

Para un valor distinto de  $n$ , y garantizar que funcione correctamente el algoritmo:

1.  $n$  debe ser una potencia de número primo.
2. Modificar la línea 9 del archivo en Python.
3. Garantizar que exista la cantidad de imágenes. Total de imágenes =  $n^2+n+1$
4. Recordar que la cantidad de símbolos por carta =  $n+1$

Ejemplo: Para  $n=7$ ; el total de cartas es 57, y la cantidad de símbolos por carta es 8

## ANEXO 3. Obtención o creación de imágenes

### Objetivo:

Brindar alternativas seguras, creativas y accesibles para obtener las 31 imágenes necesarias para generar las cartas del juego tipo Spot It.

### Opciones para los estudiantes:

#### A. Crear imágenes con IA (generación personalizada)

- Plataformas sugeridas (gratuitas o con versión gratuita):
  - Bing Image Creator: <https://www.bing.com/images/create>
  - Craiyon: <https://www.craiyon.com/>
  - Playground AI: <https://playground.com/>
- Recomendaciones:
  - Dar instrucciones sencillas y en español.
  - Guardar en formato PNG o JPG.
  - Evitar rostros de personas reales o marcas registradas.

#### B. Descargar imágenes libres de derechos

- Bancos gratuitos:
  - Pixabay: <https://pixabay.com/>
  - Unsplash: <https://unsplash.com/>
  - Pexels: <https://www.pexels.com/>
- Recomendaciones:
  - Usar imágenes vectoriales, íconos o dibujos.
  - Filtrar por “ilustraciones” o “íconos” para evitar fotos realistas.
  - Descargar imágenes cuadradas o de buena resolución.

#### C. Dibujar las imágenes a mano

- Escanear o fotografiar dibujos hechos en clase.
- Requiere ajustar tamaño y formato con ayuda del docente.

### Recomendaciones técnicas:

- Cantidad de imágenes: **31** exactamente.
- Formato: **PNG o JPG**.
- Tamaño aproximado recomendado: **100x100 píxeles** (el programa ajusta automáticamente).
- Nombres sugeridos: simbolo\_01.png, simbolo\_02.png, ... hasta simbolo\_31.png.