

# **Snake Retro - Documentación Técnica**

Justin Alonso Martínez Molina

2019027054

Kevin Jiménez Torres

2019167495

Gabriel Alonso Ruiz Corella

2015031157

Mathew Josué Carballo López

2022218443

## **Resumen**

El proyecto Snake Retro es una recreación basada en la web del clásico juego Snake, desarrollada como una herramienta educativa para demostrar principios de arquitecturas de procesamiento cooperativo y monitoreo de recursos en sistemas distribuidos. Creado para el curso de Sistemas de Procesamiento y Monitoreo de Recursos del primer semestre de 2025, este proyecto ofrece una experiencia interactiva que combina tecnologías web modernas (HTML, CSS, JavaScript) con conceptos fundamentales de computación. Además de su valor didáctico, proporciona una base modular apta para futuras ampliaciones hacia escenarios más complejos.

## **Abstract**

The Snake Retro project is a web-based recreation of the classic Snake game, developed as an educational tool to demonstrate principles of cooperative processing architectures and resource monitoring in distributed systems. Created for the I Semester 2025 course on Resource Processing and Monitoring Systems, this project offers an interactive experience that combines modern web technologies (HTML, CSS, JavaScript) with fundamental computing concepts. Beyond its educational value, it serves as a modular foundation suitable for future expansions into more complex scenarios.

## **Introducción**

El proyecto Snake Retro es una reinterpretación del juego clásico Snake desarrollada con tecnologías web estándar (HTML, CSS y JavaScript). Su objetivo principal es proporcionar una herramienta didáctica que permita explorar y comprender los principios del procesamiento cooperativo y la supervisión de recursos en sistemas distribuidos. Este proyecto fue desarrollado en el marco del curso de Sistemas de Procesamiento y Monitoreo de Recursos del primer semestre de 2025. La implementación permite a los estudiantes interactuar con una aplicación sencilla y accesible, a la vez que se introducen en conceptos clave de la programación basada en eventos y la arquitectura modular.

## Objetivos

- Recrear el juego clásico Snake con una interfaz moderna e intuitiva.
- Demostrar principios básicos de programación dirigida por eventos y gestión de recursos en entornos de navegador.
- Proporcionar una base de código modular que permita futuras extensiones enfocadas en procesamiento distribuido o cooperativo.
- Ofrecer un entorno controlado e interactivo para el análisis del uso de recursos por parte de los estudiantes.

## Metodología

### Arquitectura del sistema

El proyecto está estructurado como una aplicación web de una sola página. El frontend se encarga de toda la lógica del juego, la renderización y la interacción con el usuario. Los componentes principales son:

- **HTML (index.html):** Define la estructura de la interfaz de usuario, incluyendo menús, lienzo del juego y opciones de configuración.
- **CSS (style.css):** Proporciona el estilo visual del juego, asegurando una experiencia atractiva y accesible.
- **JavaScript (snake.js):** Contiene la lógica central del juego: movimiento de la serpiente, detección de colisiones, puntuación, gestión de sonidos y navegación de menús.
- **Recursos de audio (sounds/):** Incluye efectos de sonido que enriquecen la experiencia del usuario (movimiento, comer, choque).

### Lógica del juego

El ciclo principal del juego se gestiona mediante la función `setInterval` de JavaScript, que actualiza el estado del juego y redibuja el lienzo a intervalos regulares. La entrada del usuario se capta a través de eventos de teclado, lo que permite un control en tiempo real. El juego admite funciones como pausa, reanudación y configuración de audio, todas gestionadas mediante manipulación del DOM para mantener una experiencia fluida.

### Gestión de recursos

Aunque la implementación actual se ejecuta completamente en el navegador, la estructura modular del código está diseñada para facilitar futuras integraciones con características de monitoreo distribuido o multijugador. Esta base permite incorporar módulos externos para el seguimiento de recursos o sincronización entre usuarios.

## Pruebas y validación

El juego fue probado en diversos navegadores (Chrome, Firefox, Edge) para asegurar su compatibilidad y rendimiento. Se recogió retroalimentación de usuarios para mejorar la interfaz y accesibilidad. Las funciones de audio fueron verificadas en diferentes sistemas operativos y se implementaron mecanismos de respaldo para navegadores con restricciones de sonido.

## Resultados

Snake Retro proporciona una recreación fidedigna del juego original, mejorada con elementos de interfaz moderna y retroalimentación sonora. Entre los principales logros se destacan:

- **Diseño responsivo:** Se adapta a distintos tamaños de pantalla y dispositivos.
- **Interfaz intuitiva:** Los menús y mensajes son claros, accesibles para usuarios de cualquier nivel.
- **Código modular:** La separación entre HTML, CSS y JavaScript facilita el mantenimiento y la extensión.
- **Integración de sonido:** Mejora la experiencia de juego y el nivel de inmersión.
- **Potencial de extensión:** El diseño modular permite futuras incorporaciones como juego en red o monitoreo de recursos distribuido.

## Discusión

Durante el desarrollo se reafirmó la importancia de una arquitectura modular y de la separación de responsabilidades en aplicaciones web. Esta organización no solo mejora la mantenibilidad, sino que también allana el camino para futuras evoluciones del proyecto.

Aunque la versión actual se centra en la experiencia de un solo jugador, su arquitectura abre la posibilidad de introducir funcionalidades cooperativas o distribuidas, como el control compartido de la serpiente o el monitoreo en tiempo real del uso de recursos entre varios clientes conectados.

## Conclusiones

Snake Retro demuestra que los videojuegos clásicos pueden reutilizarse como herramientas efectivas para la enseñanza de conceptos modernos en informática. El proyecto cumple sus objetivos al ofrecer una implementación robusta y amigable del juego Snake, al mismo tiempo que sirve como plataforma para la exploración de técnicas avanzadas en sistemas distribuidos y gestión de recursos.

## Recomendaciones

- Utilizar Snake Retro como base para proyectos o tareas sobre programación dirigida por eventos y sistemas distribuidos.
- Ampliar el juego con funcionalidades multijugador o integración de APIs para monitoreo.

- Experimentar con nuevas funcionalidades, optimizar el rendimiento o analizar el uso de recursos en diferentes escenarios.
- Investigar la posibilidad de desplegar el juego en entornos distribuidos, habilitando colaboración en tiempo real y análisis compartido.

## **Referencias**

MDN Web Docs: Game development. <https://developer.mozilla.org/en-US/docs/Games>

HTML5 Canvas API. [https://developer.mozilla.org/en-US/docs/Web/API/Canvas\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API)

JavaScript Event Handling. <https://developer.mozilla.org/en-US/docs/Web/Events>