

MEMORIAS



CURSO:
Diseño de objetos interactivos
con DescartesJs e IA

Carlos Alberto Rojas Hincapié



MEMORIAS



CURSO:

Diseño de objetos interactivos
con DescartesJS e IA

Tutores

Juan Guillermo Rivera Berrío
Jesús Manuel Muñoz Calle



Carlos Alberto Rojas Hincapié
Red Educativa Digital Descartes, Colombia
2025

Título de la obra:
MEMORIAS

CURSO:

Diseño de objetos interactivos con DescartesJS e IA

Carlos Alberto Rojas Hincapié

Tutores

Juan Guillermo Rivera Berrío

Jesús Manuel Muñoz Calle

Código JavaScript para el libro: [Joel Espinosa Longi](#), [IMATE](#), UNAM.

Recursos interactivos: [DescartesJS](#)

Fuentes: [Lato](#) y [UbuntuMono](#)

DATOS DE LA EDICIÓN

ISBN: 978-84-10368-26-2



Esta obra está bajo una licencia [Creative Commons 4.0 internacional: Reconocimiento-No Comercial-Compartir Igual](#).



Puedes descargar el libro en formato pdf:

[Descargar PDF](#)



Tabla de contenido

Prefacio	1
----------	---

Sesión 1

1. Diseño de OIA con DescartesJS e IA	4
1.1 Introducción	5
1.1.1 Objetos Interactivos	5
1.1.2 Objetos interactivos con DescartesJS	7
1.2 Otras herramientas de autor	9
1.2.1 Herramientas	9
1.2.2 Herramientas IA	15
1.3 Plantillas	21
1.3.1  Plantillas DescartesJS	21
1.3.2  Plantillas con IA	23
1.4  Video - Sesión 1ª	25

Sesión 2

2. El editor de configuraciones de DescartesJS	26
2.1 El editor de configuraciones de DescartesJS	27
2.2 Estructura del editor de configuraciones	29
2.2.1 ¿Qué se configura en el apartado escena?	30
2.2.2 ¿Cómo se configuran los espacios en Descartes?	31
2.3 Incorporación de OIA diseñados con IA en un espacio cartesiano	37
2.4  Video - Sesión 2ª	42

Sesión 3

3. Diseño de OIA con DescartesJS e IA	44
3.1 Los formatos de imagen	45
3.2 Imágenes en DescartesJS	48
3.3 Presentadores de diapositivas	51
3.4  Video - Sesión 3ª	58

Sesión 4

4. Teorema del Programa Estructurado	60
4.1 Nivel 1 - Introducción	61
4.2 Nivel 2 - Profundización	64
4.3 Estructuras secuenciales y selectivas	67
4.3.1 Estructuras secuenciales	67
4.3.2 Estructuras selectivas	71
4.4  Video - Sesión 4ª	77

Sesión 5

5. Estructuras iterativas	80
5.1 Nivel 1 - Introducción	81
5.2 Nivel 2 - Estructuras iterativas y vectores	84
5.3 Nivel 3 - Lectura de datos externos	87
5.4  Video - Sesión 5ª	90

Sesión 6

6. Los controles del editor DescartesJS	92
6.1 Nivel 1 - Introducción	93

6.2 Nivel 2 - Controles tipo pulsador y tipo barra	99
6.3 Nivel 3 - Controles tipo deslizador y casilla de verificación	101
6.4  Video - Sesión 6ª	104

Sesión 7

7. Audios y videos	106
7.1 Nivel 1 - Introducción	107
7.2 Nivel 2 - Diseño de vídeos interactivos con el editor DescartesJS	111
7.3 Nivel 3 Diseño de vídeos interactivos con IA	115
7.4  Video - Sesión 7ª	118

Sesión 8

8. Familias de gráficos	120
8.1 Nivel 1 - Introducción	121
8.2 Nivel 2 Diseño de secuencias en DescartesJS	125
8.3 Nivel 3 Diseño de secuencias con herramientas de IA	129
8.4  Video - Sesión 8ª	132

9. Tareas y Proyectos de aula	134
9.1 Actividades (tareas)	135
9.2 Proyectos	151

Prefacio

El nuevo [Marco de Referencia para la Competencia Digital Docente](#) recoge, en el área 2 dedicada a contenidos digitales, entre sus elementos específicos:

- "Utilización de herramientas de autor para la modificación, diseño y creación de contenidos educativos digitales nuevos o derivados de otros contenidos..."
- "Aplicación de criterios de calidad de los contenidos educativos digitales para su adaptación al alumnado con el que se trabaja y a los fines perseguidos".
- "Respeto de los derechos de autor y de las limitaciones de la propiedad intelectual aplicadas al ámbito educativo para la reutilización y la creación de contenidos..."

Pues bien, desde RED Descartes te ofrecemos la posibilidad de abordar estos elementos específicos, como mínimo, aprendiendo el manejo de la herramienta de autor DescartesJS, un software libre y multipropósito con el que podrás crear tus propios contenidos interactivos para cualquier materia, respetando los derechos de autor y compartiéndolos en entornos seguros con una licencia Creative-Commons. Pero, además, conocerás y podrás aprovechar el enorme potencial de la Inteligencia Artificial en el ámbito educativo usando herramientas que te facilitarán la práctica docente hasta límites insospechados.

Queda abierto, por tanto, el plazo de inscripción gratuita a la I Edición del Curso para el Diseño de objetos interactivos con IA y DescartesJS, una acción formativa que se enmarca en el programa de Educación Abierta desarrollado entre redes docentes de Colombia, México y España, fundamentalmente, aunque contamos con la participación de profesorado de otros países de habla hispana, portuguesa e inglesa.

Un curso que tiene como objetivo principal diseñar recursos educativos abiertos en formato de escenas interactivas con la herramienta de autor Descartes JS y en formato web con herramientas de Inteligencia Artificial (WebSim, Claude 3.7, vo de Vercel, Pollinations, entre otras), que permitan poner de manifiesto que es posible dar una respuesta positiva y asequible a los retos educativos intrínsecos al paradigma educativo emergente.

Está dirigido a docentes de cualquier etapa educativa, infantil, primaria, educación secundaria obligatoria, bachillerato, formación profesional, enseñanzas de régimen especial y universidad, y de cualquier materia o especialidad, en activo o no, así como a profesionales vinculados a la educación o formación y a diseñadores y desarrolladores interesados en objetos interactivos, utilizando una metodología activa, concretamente Clase Invertida y Aprendizaje Basado en Proyectos (Tareas), pues desde la primera sesión cada participante comenzará a diseñar y editar su primer recurso interactivo, recibiendo sesiones quincenales por videoconferencia, que serán grabadas y compartidas con todos los participantes y asesorados por docentes de las redes mencionadas.

- [Descripción detallada de cada sesión](#)

Recursos y tutoriales previos:

1. [Descarga el software libre DescartesJS](#) para tu sistema operativo
2. [Tutorial sobre DescartesJS - Nivel I](#)
3. [Herramientas de Inteligencia Artificial para la enseñanza](#)
4. [Lista en Youtube con vídeos de apoyo.](#)



Sesión 1

Diseño de OIA con
DescartesJS e IA

1.1 Introducción

1.1.1 Objetos Interactivos

¿Qué es un objeto interactivo de aprendizaje?

Definición:



Es importante comprender lo que entendemos por un objeto interactivo de aprendizaje - OIA -, pues de ello depende reconocer la importancia que tiene en un proceso de enseñanza aprendizaje, pues no se trata de los “objetos virtuales de aprendizaje” - OVA -, que podrían limitarse a una simple imagen, animación o vídeo.

Los OIA son representaciones más efectivas porque no sólo explican sino que permiten inferir desde el objeto representado. La interacción sujeto - objeto, al poder cambiar atributos del objeto por parte del sujeto, permite un mayor conocimiento del objeto.



Para complementar y no extendernos en la argumentación, recomendamos la lectura del siguiente artículo:

MODELOS DE INTERVENCIÓN CON DESCARTES.¹



S.A.E.M.
THALES

Córdoba 10, 11 y 12
de Septiembre de 2010

MODELOS DE INTERVENCIÓN CON *DESCARTES*

Juan Guillermo Rivera Berrío, *Proyecto Descartes (ITE del Ministerio de Educación) e Instituto Tecnológico Metropolitano de Medellín (Colombia)*

José Román Galo Sánchez, *Proyecto Descartes (ITE) y Universidad de Córdoba*

José Luis Alcón Camas, *Proyecto Descartes (ITE) e Instituto Baelo Claudia de Tarifa (Cádiz)*

RESUMEN.

Los modelos didácticos que desarrollan conceptos científicos son un canal usual de explicación y transmisión docente. Con ellos se busca motivar al discente para que infiera y deduzca propiedades, pero los soportes tradicionales (pizarra o papel) suelen constreñir este objetivo. Son los actuales sistemas interactivos de simulación los que permiten elaborar modelos pedagógicos de intervención en los que el aprendizaje significativo se manifiesta de manera natural, surgiendo espontáneamente la conjugación de la labor descubridora e investigadora. En esta comunicación se muestra cómo *Descartes* es un mediador virtual que sirve de soporte básico en la elaboración de modelos de intervención. Esta mediación se ejemplifica aquí a través de las percepciones de la Gestalt.

Nivel educativo: Secundaria y Universidad.

1. MODELOS EN LOS PROCESOS DE ENSEÑANZA APRENDIZAJE.

Tradicionalmente en nuestra labor docente hacemos uso de diferentes modelos educativos que buscan la explicación de conceptos bien matemáticos, físicos, lógicos o de otra índole. A veces recurrimos a metáforas o analogías desde asunciones que siendo conscientes a priori que son falsas (modelos del

¹ September 2010, Conference: XIII Congreso sobre enseñanza y aprendizaje de las Matemáticas (XIII CEAM)At: Córdoba (España). Authors: Juan Guillermo Rivera Berrío, Galo Sánchez José R. y José L. Alcón Camas. [Artículo](#).

1.1.2 Objetos interactivos con DescartesJS

¿DescartesJS?

Mini historia: El Proyecto Descartes nace a finales del siglo XX como una herramienta de creación de escenas interactivas de matemáticas, aprovechando el lenguaje de programación Java y permitiendo su visualización en páginas web.

Alrededor del año 2013, con la desaparición del soporte para Java en los navegadores modernos, se decidió migrar el proyecto a JavaScript y HTML5, dando lugar a DescartesJS. El editor ha sido desarrollado en la Universidad Nacional Autónoma de México (UNAM), manteniendo la filosofía del software original pero sin la dependencia de complementos externos.

Con DescartesJS hemos diseñado una gran cantidad de OIA, con la posibilidad de descargar e intervenir; entre ellos, libros interactivos de aprendizaje - LIA -, vídeos interactivos de aprendizaje - VIA -, Unidades Interactivas de Aprendizaje, etc.



Figura 1.1. [Red Educativa Descartes](#).

A continuación, presentamos dos repositorios en los que se alojan estos OIA:

- ✦ [Proyecto Prometeo](#) (México)
- ✦ [Proyecto Descartes](#) (España).

Al lado derecho, puedes interactuar con una escena diseñada con DescartesJS, la cual puedes descargar en un archivo zip [aquí](#) 



Versión: 1.4.2
descartes-min.js: 2024-31-10_1
NW.js: 0.88.0

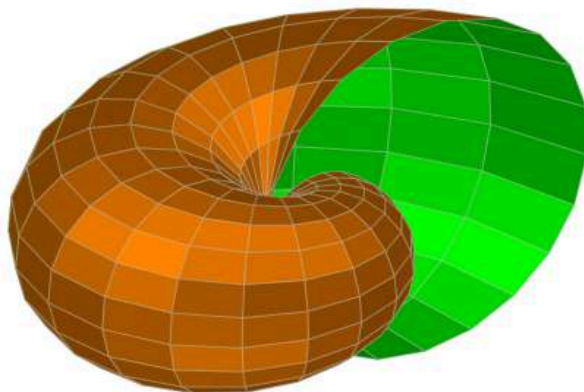
El editor Descartes es un programa de código abierto y gratuito, programado por Joel Espinosa Longi ([Instituto de Matemáticas, UNAM](#)), basado en el [editor Descartes](#).



¿Cómo instalo el editor DescartesJS?

Instalación: Descartes es una herramienta de autor diseñada para facilitar la creación de materiales educativos interactivos, de uso libre y gratuito, dirigida a profesores y estudiantes de toda Iberoamérica.

La versión actual es la 1.4, pero en permanente actualización y mejora. En este [enlace](#), puedes descargar el editor DescartesJS, disponible para Windows, Mac OS y Linux. Solo tienes que descargar el instalador y ejecutarlo, según el sistema operativo.



ver Sólido ▾

t  2.00

a  1.50

1.2 Otras herramientas de autor

1.2.1 Herramientas



H5P



Descripción:

H5P (HTML5 Package) es un framework de código abierto gratuito para crear, compartir y reutilizar contenido HTML5 interactivo. Permite a los usuarios crear contenido rico como presentaciones interactivas, videos, cuestionarios y más, directamente en sus navegadores sin necesidad de habilidades de programación. La herramienta es ampliamente compatible con varios sistemas de gestión del aprendizaje (LMS).

Importancia:

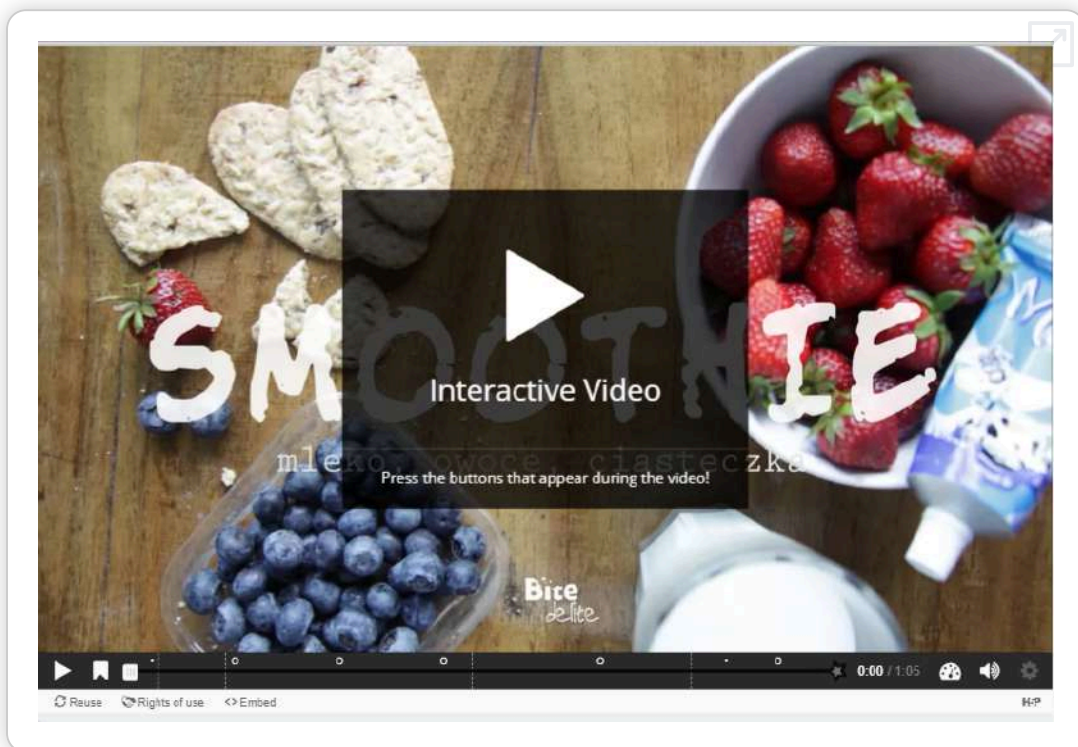
H5P democratiza la creación de contenido interactivo, haciéndola accesible a educadores y creadores sin habilidades técnicas especializadas. Su compatibilidad con los LMS populares como Moodle y Canvas facilita la integración en los flujos de trabajo educativos existentes. Facilita experiencias de aprendizaje más atractivas y efectivas para los estudiantes.



Ejemplo, videos interactivos:

Los videos pueden enriquecerse con actividades interactivas como explicaciones, imágenes adicionales, tablas, preguntas para completar espacios en blanco y preguntas de opción múltiple.

Las preguntas de prueba admiten la adaptabilidad, lo que significa que puede saltar a otra parte del video según la entrada del usuario. Se pueden agregar resúmenes interactivos al final del video. Los videos interactivos se crean y editan utilizando la herramienta de creación de H5P en un navegador web estándar.



En <https://h5p.org/> puedes interactuar con el video interactivo. En este curso, veremos cómo crear videos interactivos con DescartesJS y con IA.



Genially es una plataforma en línea para crear presentaciones interactivas, infografías, juegos y más. Fue fundada en España en 2015.

Se ha popularizado por su facilidad de uso y la capacidad de crear contenido visualmente atractivo sin necesidad de conocimientos técnicos avanzados en diseño.

Importancia:

Ha democratizado el diseño interactivo, permitiendo a educadores, empresas y particulares crear presentaciones y materiales didácticos más atractivos.

Esto ha mejorado la comunicación y el aprendizaje en diversos contextos, desde el aula hasta el marketing digital.

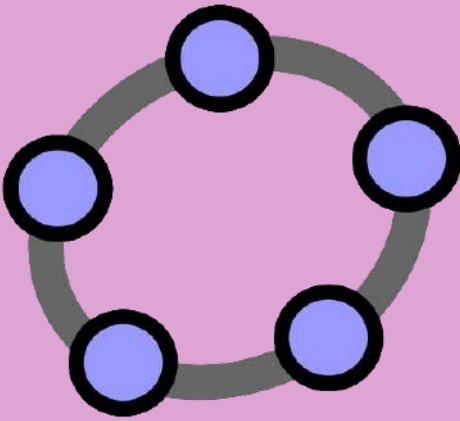
En la 3º sesión de este curso, crearás tus presentaciones interactivas con DescartesJS y con IA.



Ejemplo, presentaciones interactivas:

Con Genially puedes crear presentaciones interactivas de forma muy fácil. La presentación siguiente fue realizada con esta herramienta:





GeoGebra

Descripción:

GeoGebra ofrece una interfaz dinámica que conecta geometría, álgebra, hojas de cálculo, gráficos, estadística y cálculo en un único paquete fácil de usar. Esto permite a los usuarios explorar conceptos matemáticos de forma interactiva, visualizando las relaciones entre diferentes ramas de las matemáticas. Es ampliamente utilizado en la educación para demostrar conceptos que serían difíciles de ilustrar estáticamente. ([Instalación de GeoGebra](#))

Importancia:



Esta característica facilita la comprensión profunda de los principios matemáticos. Al permitir a los estudiantes manipular objetos matemáticos y observar los resultados de inmediato, GeoGebra mejora el aprendizaje exploratorio. Además, su versatilidad lo convierte en una herramienta valiosa para profesores en diversos niveles educativos.

Ejemplo, la ilusión del cuadrado palpitante:

Ilusión de Misha Pavel. Los seres humanos yerran constantemente en su percepción del movimiento rotatorio vistos a través de una abertura. Una abertura se constituye en una restricción para la percepción visual, la cual se convierte en parte del contexto del objeto percibido. A veces, este nuevo elemento del contexto es preferido por nuestro sistema visual, en tanto que es rígido y demanda menor esfuerzo neuronal. De esta forma, prima la restricción sobre los otros elementos en movimiento. Según Shiffrar & Pavel: "The visual system has a preference to interpret the image as containing a single, rigid object".



Ver objeto interactivo en la página de [GeoGebra](#).

1.2.2 Herramientas IA

WebSim



Descripción:

WebSim AI es una herramienta de creación de sitios web de vanguardia que aprovecha modelos avanzados de IA como Claude 3.7 Sonnet, Gemini Flash 2.0, DeepSeek R1 y o1 mini para revolucionar el desarrollo web.

Permite a usuarios de todos los niveles de habilidad crear sitios web y aplicaciones web funcionales y dinámicas simplemente describiendo su visión en lenguaje natural o ingresando una URL. WebSim AI transforma el diseño web estático en un entorno interactivo y exploratorio donde las ideas pueden ser prototipadas rápidamente y llevadas a la vida sin conocimientos de codificación. Cuando se le da una solicitud o URL, la IA la analiza y crea un sitio web o aplicación web completamente funcional, completo con texto, imágenes y elementos interactivos. (<https://websim.com/>)

Importancia:

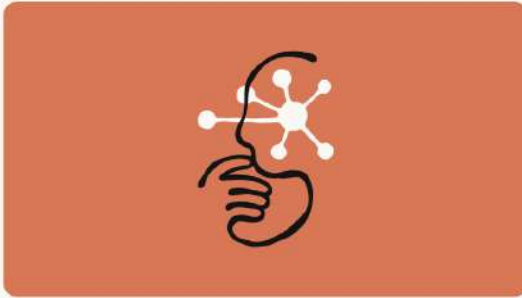
WebSim AI ofrece numerosos beneficios a los usuarios en diversos campos. Para diseñadores y desarrolladores, sirve como una poderosa herramienta de ideación y prototipado, permitiendo la visualización rápida de conceptos sin la codificación que consume tiempo.

Los emprendedores y los especialistas en marketing pueden crear y probar rápidamente diferentes iteraciones de sitios web para recopilar comentarios de los usuarios. Los educadores pueden usarlo para crear entornos de aprendizaje interactivos. La plataforma democratiza el desarrollo web, permitiendo a los usuarios no técnicos dar vida a sus ideas. Su flexibilidad y facilidad de uso fomentan la creatividad y la experimentación, lo que puede llevar a experiencias y aplicaciones web innovadoras que podrían no haberse concebido a través de métodos de desarrollo tradicionales.



Claude 3.7 Sonnet and Claude Code

Feb 25, 2025 • 5 min read



Descripción:

Anthropic lanzó su modelo de IA más inteligente hasta la fecha y el primer modelo de razonamiento híbrido del mercado: Claude 3.7 Sonnet. La parte híbrida

significa que el modelo funciona tanto como modelo de razonamiento como modelo de lenguaje extenso (LLM).

Si bien OpenAI anunció recientemente que GPT-5 será un modelo unificado, Anthropic ya presentó Claude 3.7 Sonnet, que es capaz tanto de brindar respuestas rápidas como de razonamiento más profundo, adelantándose en este enfoque particular para el desarrollo de IA.

Importancia:

Este nuevo modelo puede “pensar” en las preguntas durante el tiempo que los usuarios se lo pidan, por lo que dependiendo del tiempo que considere las cosas, sus respuestas podrían ser muy diferentes. Claude 3.7 Sonnet también puede crear aplicaciones complejas con una sola indicación y, con la introducción de un nuevo producto llamado Claude Code, los desarrolladores ahora pueden asignar tareas de ingeniería importantes a Claude directamente desde su terminal ([Medium](#)).



Descripción:

V0 de Vercel (<https://v0.dev/>) es una herramienta de inteligencia artificial generativa que optimiza el desarrollo de aplicaciones web al transformar la información del usuario en código listo para usar. Es compatible con varios marcos como React, Vue, Svelte y HTML

con CSS, lo que la hace versátil para diferentes requisitos de proyectos. La herramienta está diseñada para ser accesible incluso para aquellos con experiencia limitada en codificación. En palabras simples, es su asistente personal que sabe cómo escribir código e implementarlo en producción. V0 de Vercel es un punto de inflexión en lo que respecta a la creación de interfaces de usuario basadas en IA. Con su enfoque innovador, simplifica todo el proceso y lo hace accesible para desarrolladores de todos los niveles.

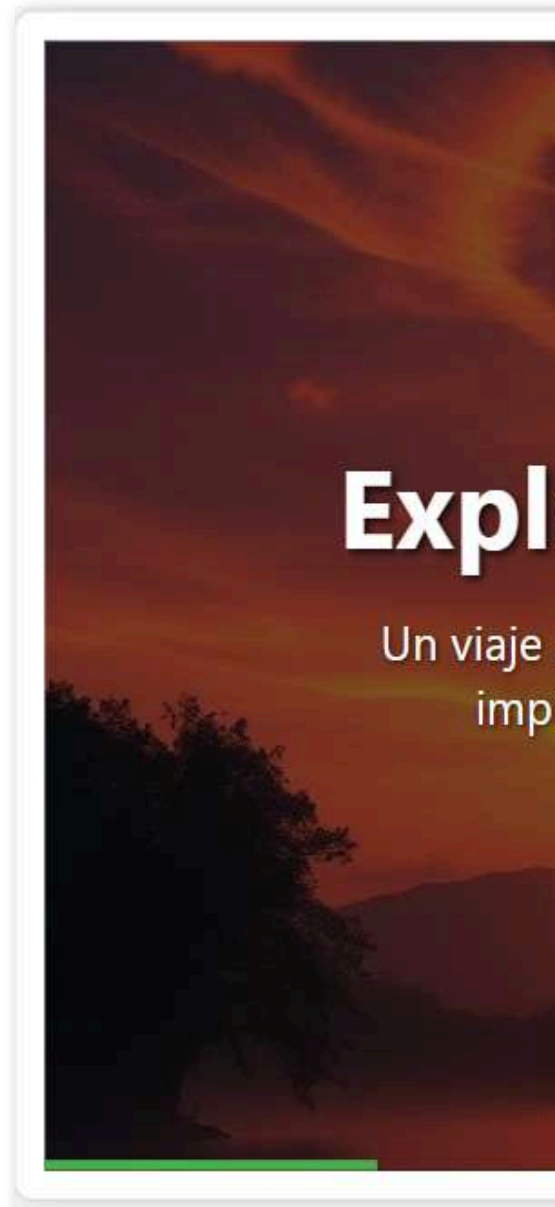
Importancia:

La versión V0 de Vercel utiliza una interfaz basada en chat donde los usuarios pueden ingresar indicaciones o cargar archivos para describir la aplicación web que desean. Esta interfaz permite a los usuarios interactuar con la IA de manera conversacional, lo que la hace intuitiva y fácil de usar. La interfaz es similar a ChatGPT, por lo que es fácil de usar y comprender ([Medium](#)).

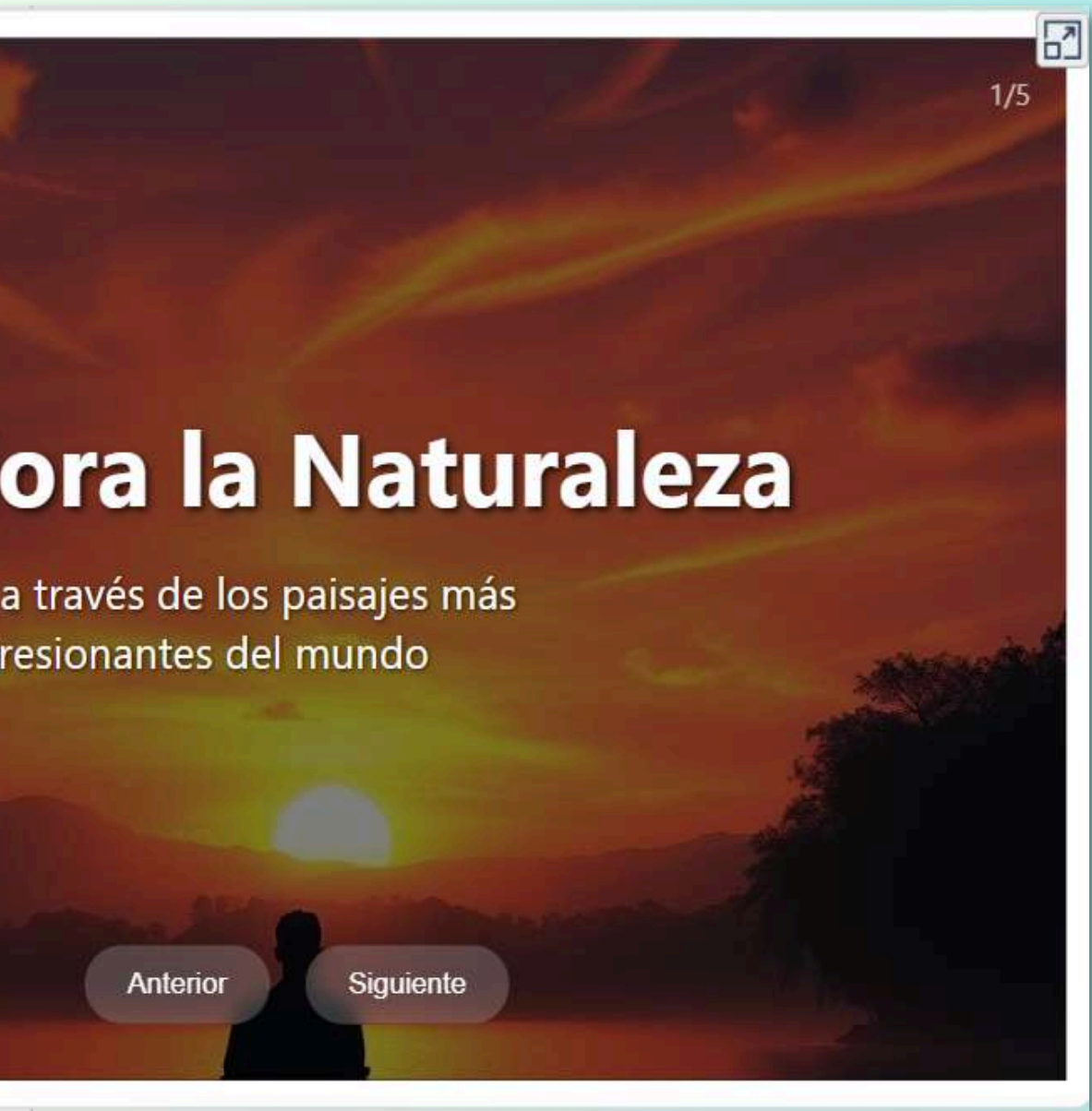
El presentador de la derecha, se obtuvo con este prompt:



"Haz una presentación de 5 diapositivas en HTML5 (CSS+JS+HTML) con imágenes generadas por pollinations.ai. Incluye botones de avance y retroceso y, además, transiciones impactantes y variadas entre diapositivas".



En esencia, v0 está diseñado para agilizar el proceso de creación de interfaces de usuario, desde la idea inicial hasta la implementación, utilizando la potencia de la inteligencia artificial.



En resumen, v0 es una herramienta de IA que actúa como un "programador de pares" para interfaces de usuario, ayudando a generar código y diseño a partir de descripciones textuales.

1.3 Plantillas

1.3.1 Plantillas DescartesJS



Descripción:

El Plantillero DescartesJS es una herramienta que permite a cualquier usuario, con mínimos conocimientos de los entornos digitales, crear contenidos interactivos de acuerdo a sus necesidades. Está dirigido, especialmente, para profesores que incorporan en su quehacer docente las Tecnologías de Información y Comunicación – TIC –, de tal forma que a

través de intervenciones simples de las plantillas que se presentan, el profesor pueda enriquecer los contenidos de las asignaturas que desarrolla con sus estudiantes.

En 2012 se diseñaron las primeras plantillas con el editor Descartes de ese entonces, que obligaba a tener instalada la máquina Java en los equipos. Ahora, estamos entregando a la comunidad académica un nuevo proyecto de plantillas en formato JavaScript – JS –, que permite su interacción, tanto en ordenadores de escritorio y portátiles como en dispositivos móviles.

Importancia:

En esta segunda edición del libro [Plantillas DescartesJS](#) se presentan 36 nuevas plantillas, 27 de ellas diseñadas para tiempos de pandemia. El libro está dirigido a profesores que incorporan en su quehacer docente las Tecnologías de la Información y Comunicación – TIC –, de tal forma que a través de intervenciones simples pueda enriquecer los contenidos de las asignaturas que desarrolla con sus estudiantes. Se incluye el documento PDF del libro.

Ejemplo, secuencias temporales:

Un ejemplo de plantilla que diseñaremos en este curso.

SECUENCIAS TEMPORALES

**Arrastra la imágenes a los cajones, en orden temporal.
Procura no dejar imágenes montadas.**

1	2	3	4



Puedes interactuar con la escena diseñada con DescartesJS, la cual puedes descargar en un archivo zip [aquí](#) 

1.3.2 📄 Plantillas con IA



Descripción:

En un mundo donde la tecnología avanza a pasos agigantados, la inteligencia artificial (IA) se ha convertido en una herramienta esencial para la educación y la creación de contenidos. Este [libro](#), explora cómo la IA puede transformar y enriquecer el proceso de aprendizaje. A lo largo de las páginas, el lector encontrará una valiosa colección de

plantillas y herramientas, diseñadas para facilitar la creación de libros, juegos, presentaciones y otras actividades académicas enriquecidas por la IA. Desde sencillas sopas de letras hasta complejos cuestionarios y juegos interactivos, cada una de estas plantillas ha sido cuidadosamente desarrollada para brindar a educadores, diseñadores y desarrolladores una base sólida sobre la cual construir sus propias propuestas creativas.

📌 Importancia:

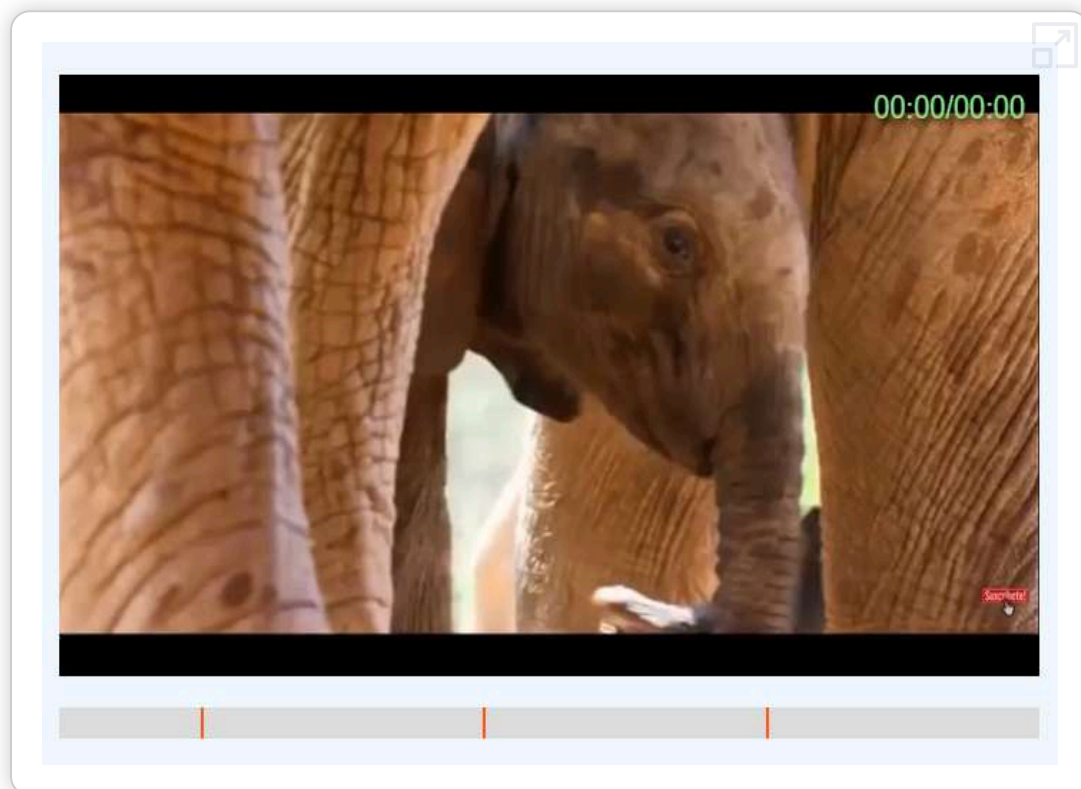
Un aspecto destacable de este trabajo es la manera en que los autores han logrado combinar eficazmente las capacidades de la inteligencia artificial con los principios del diseño instruccional y las mejores prácticas en el campo de la educación digital.

Cada plantilla no solo ofrece una funcionalidad técnica avanzada, sino que también está alineada con enfoques pedagógicos probados, lo que garantiza una experiencia de aprendizaje enriquecedora y significativa para los usuarios finales.

Esperamos que este libro se convierta en una herramienta indispensable para todos aquellos interesados en explorar las posibilidades que ofrece la integración de la IA en el diseño y desarrollo de recursos educativos. Que disfruten de este viaje de descubrimiento y que encuentren en estas páginas la inspiración y los conocimientos necesarios para llevar sus proyectos al siguiente nivel.

Ejemplo, Videos interactivos:

Un ejemplo que diseñaremos en este curso.



1.4 🎬 Video - Sesión 1ª

📌 *Diseño de objetos interactivos con DescartesJS e IA*



📁 Tarea N°1

✂️ Elaborar un cuestionario generado con una herramienta IA.

Ver una posible

✓ [Solución](#)

Sesion 1. 📄 ¡Que aprendiste!



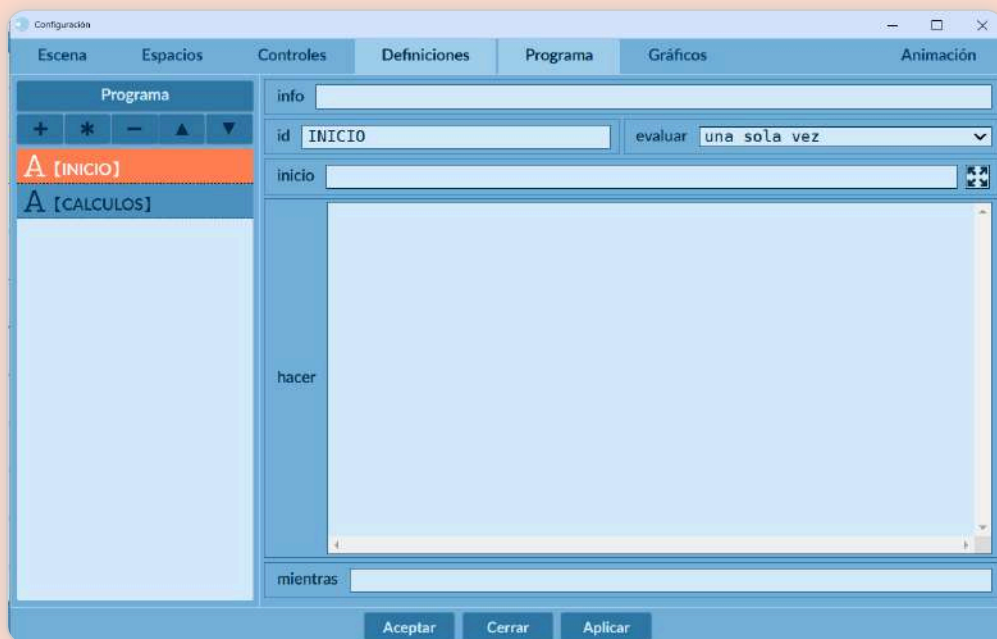
Sesión 2

**El editor de configuraciones
de DescartesJS**

2.1 El editor de configuraciones de DescartesJS

¿Qué es el editor de configuraciones?

Acceso: En el editor de DescartesJS, haces clic en  para abrir el editor de configuraciones.



Definición:

El editor de configuraciones es una ventana que acompaña a la escena cuando se está creando o editando y es donde se realiza la programación de un recurso. Contiene las herramientas necesarias para crear y modificar una escena. En la siguiente página se encuentra la documentación completa del editor de DescartesJS.

Todos los objetos interactivos que diseñaremos en este curso, con el editor DescartesJS, se harán a través de este editor de configuraciones.



DESCARTES
matemáticas interactivas

Documentación de Descartes JS

Alejandro Radillo Díaz

José Luis Abreu León

Joel Espinosa Longi

24 de julio de 2025

2.2 Estructura del editor de configuraciones

¿Cómo es el editor de configuraciones?

El editor dispone de un menú superior con los principales apartados a través de los cuales se configura o programa cada escena. Cada uno de los selectores tiene su propio espacio de configuración que permiten introducir los diferentes elementos de las escenas y realizar la programación de la misma.

El editor de configuraciones de una escena sólo se puede abrir desde el editor de escenas de Descartes, cuya descarga e instalación se explicó en la sesión 1.

 **Guía en video.**



El menú de configuración

2.2.1 ¿Qué se configura en el apartado escena?

Permite configurar aspectos generales de toda la escena, tales como sus dimensiones, los bordes de la escena como zonas para introducir controles, la presencia o no de una serie de controles por defecto, la imagen que se presenta durante la carga de la escena, el uso del punto o la coma decimal, etc.

Un parámetro muy interesante de este selectorel el de expandir escena, ya que en su opción escalar permite hacer que la escena se ajuste al tamaño de su contenedor, haciendo por tanto que esta sea adaptable o "responsive".

Configuración

Escena Espacios Controles Definiciones Programa Gráficos Animación

Título de la página: TITULO

ancho: 970 alto: 550

botón créditos ☐ botón config ☐ botón inicio ☐ botón limpiar ☐

filas al norte: 0 filas al sur: 0

ancho oeste: 150 ancho este: 150 alto filas: 40

signo decimal: . idioma: español mostrar región exterior ☐ expandir escena: ☐

Imagen del cargador:

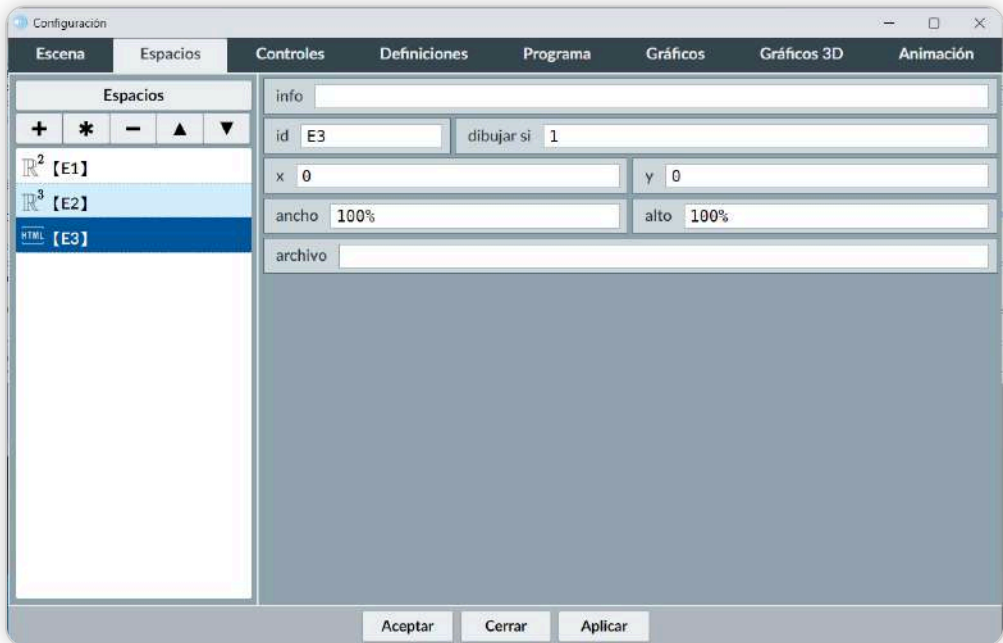
Aceptar Cerrar Aplicar

2.2.2 ¿Cómo se configuran los espacios en Descartes?

Descripción:

Para entender que son los espacios utilizaremos el siguiente símil. Imaginemos que cada espacio de una escena es una hoja de papel. Podemos tener hojas de papel de diferentes tamaños, colocadas unas encima de otras y cada hoja de papel puede tener escritos diferentes contenidos. Si tenemos apiladas un montón de hojas de papel (espacios), desde arriba veremos las hojas que quedan encima y que no están cubiertas por otras hojas. Si una hoja es transparente o semitransparente, dejará ver total o parcialmente la hoja que hay debajo.

En la siguiente imagen se muestra una captura de pantalla en la que se han incluido tres espacios diferentes de Descartes.



Tipos de espacios:

Son tres los espacios que podemos usar con DescartesJS: R^2 o bidimensional, R^3 o tridimensional y los de tipo HTMLIframe que permiten incorporar cualquier tipo de página web.

Cuando se ejecuta el Editor de DescartesJS, por defecto, siempre se muestra un espacio cartesiano bidimensional. Este espacio se identifica (en el selector Espacios) normalmente como E1.

Guía en video.

El siguiente vídeo muestra la forma de insertar y configurar de forma rápida los diferentes espacios en DescartesJS.



Mi primer ejercicio

Mi primer Objeto Interactivo con DescartesJS

Esto se aprende es "haciendo", así que vamos a diseñar nuestra primera escena con el editor DescartesJS.

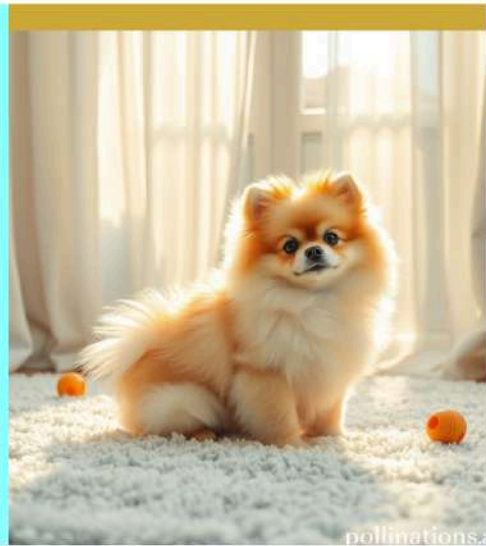
El ejercicio consiste en diseñar una escena con tres espacios 2D, como la que se muestra en la imagen, incluyendo texto e imagen.

El primer espacio es el espacio por defecto E1, agregando dos espacios adicionales, que ocupan un 50% del lienzo E1.



¡He aquí un lindo perrito!

Se trata de un perro pomerania, una raza pequeña ideal como perro de compañía



Realiza el ejercicio y lo guardas con el nombre ejercicio1, DescartesJS pone, automáticamente, la extensión html, para que puede ver el ejercicio en cualquier navegador.

¿Cómo diseñar la escena?

Es importante que se cree un directorio en el cual vamos a guardar nuestros ejercicios, tareas y proyectos.

Guía en video.

En el siguiente video se explica cómo diseñar la escena anterior.



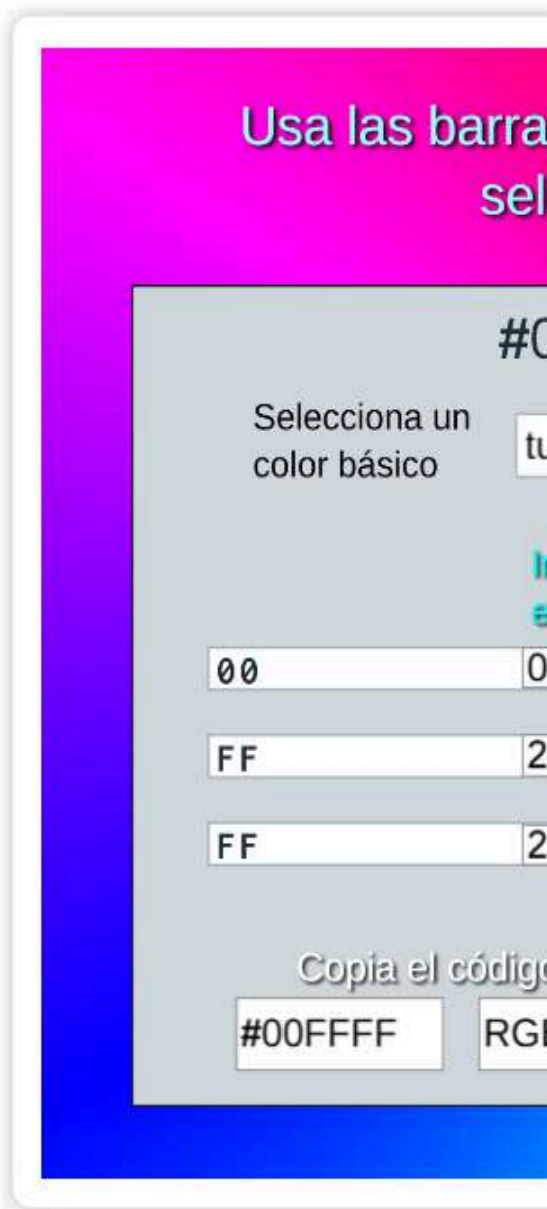
Uso de colores

Uno de los parámetros más utilizados es el de colores. Está presente en la mayoría de los elementos.

A continuación, hemos incluido un interactivo que permite practicar la forma de seleccionar los colores que afectan a los elementos. Además, desde este menú también se configura el grado de transparencia que se le quiera dar.



En las últimas versiones de Descartes se ha incluido la posibilidad de utilizar gradientes de colores, imágenes patrón y la opción de elegir colores con un "Cuentagotas".



¿Cuáles son los códigos de color de HTML? Los códigos de colores son formas de representar los colores que vemos cada día en un formato que un ordenador puede interpretar y mostrar. Comúnmente utilizado en sitios web y otras aplicaciones de software, hay una gran variedad de formatos.

s para encontrar colores o los menú para
seleccionar uno básico o especial

00FFFF

RGB(0, 255, 255)

urquesa



Selecciona un
color especial

Colores especiales



ingresa los colores
en valor decimal

55

55

o (CTRL+C)

B(0, 255, 255)



Mueve la barras para
cambiar los colores en
valores hexadecimales



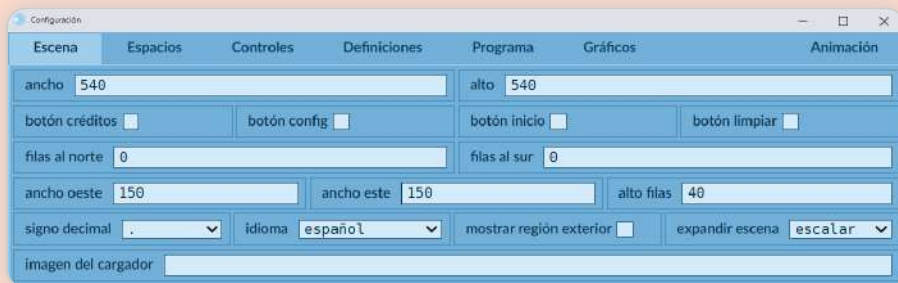
Diseño: Juan Gmo. Rivera Berrio

2.3 Incorporación de OIA diseñados con IA en un espacio cartesiano

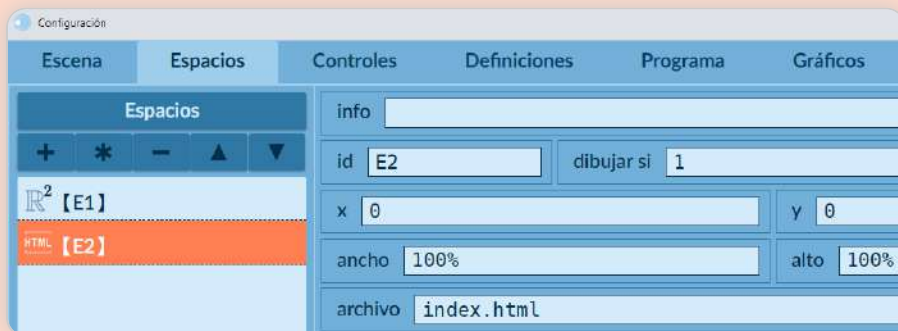
Objetos adaptables

Son objetos adaptables aquellos que se ajustan al tamaño de la pantalla. Hay objetos que están programados de serie con esta naturaleza, pero otros no.

Paso 1: Creamos una escena de las dimensiones que deseemos con la opción escalar en expandir escena.



Paso 2: Creamos un espacio tipo HTMLiframe que cargue el archivo del control deslizante, guardamos con el nombre que queramos y... eso es todo.



Ahora el objeto interactivo que hayamos introducido de la forma indicada en Descartes será "responsive" o adaptable.

OIA en local

Lo ideal, en el diseño de objetos interactivos, es usar todos los contenidos en local, pues de esa forma se garantiza un 100% de portabilidad y uso por parte de usuarios con problemas de conectividad. La mayoría de los contenidos digitales interactivos de la Red Educativa Digital Descartes, han sido diseñados para ser descargados y usados en local.

Al lado derecho, presentamos un ejemplo que incluye un espacio HTMLIframe, en el cual se carga un cuestionario diseñado por una IA y almacenado en un directorio de esta presentación (en local). Para verlo en pantalla completa, hacer clic en este [enlace](#)

Al lado derecho hemos creado un espacio HTMLIframe con un OIA correspondiente a un cuestionario diseñado con IA

Cuestionario de Números Enteros

Aciertos: 0 Fallos: 0

1/10

¿Qué son los números enteros?

Solo los números positivos

Los números positivos, negativos y el cero

Solo los números negativos

Los números decimales

Anterior

Siguiente

En este curso, vamos a aprender a diseñar herramientas que usan IA generativas como la usada en el puzle giratorio.

OIA en la red

Pese a nuestros propósitos de usar solo contenidos digitales en local, hay excepciones que obligan a los usuarios contar con conectividad a Internet.

Una de esas excepciones es la IA misma, pues el uso de algunas herramientas con ChatGPT, Gemini, Notebook, DALLE-3 y un largo etcétera, obligan a estar conectado a Internet.

En la ventana derecha, hemos puesto una escena de dos espacios, siendo uno de ellos un HTMLiframe que carga un archivo de la red (verlo [aquí](#) en pantalla completa).

Este archivo hace parte de un caja de herramientas publicada en este [enlace](#).



La herramienta utilizada es un puzle giratorio que genera imágenes desde la plataforma Pollinations AI.

Al lado derecho, presentamos un ejemplo que incluye un espacio HTMLiframe, en el cual se carga un cuestionario diseñado por una IA y almacenado en un directorio de esta presentación (en local). Para verlo en pantalla completa, hacer clic en el enlace anterior.

En este curso, vamos a aprender a diseñar herramientas que usan IA generativas como la usada en el puzle giratorio.

Al lado derecho hemos creado un espacio **HTMLiframe** con un OIA correspondiente a un cuestionario diseñado con IA y con carga de imágenes desde la Red.

Puzzle Giratorio - Imágenes con Pollinations

[Generar](#)[3x3](#)[4x4](#)[5x5](#)[6x6](#)[Reiniciar Puzzle](#)[⬇ Descargar Puzzle](#)

Diseñado por Juan Gmo. Rivera Berrio con WebSim y v0 de Vercel

Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)

Proyecto 1 - Generador de imágenes con IA (Parte 1)



Observa y sigue las instrucciones del video en la ventana derecha. Se trata de un escena interactiva en la que incluimos dos espacios, uno de ellos con imágenes generadas por IA. Es un primer paso para el primer proyecto a diseñar en este curso.

 [Guía en video. \(Parte 1\)](#)



2.4 🎬 Video - Sesión 2ª

📌 Diseño de objetos interactivos con DescartesJS e IA



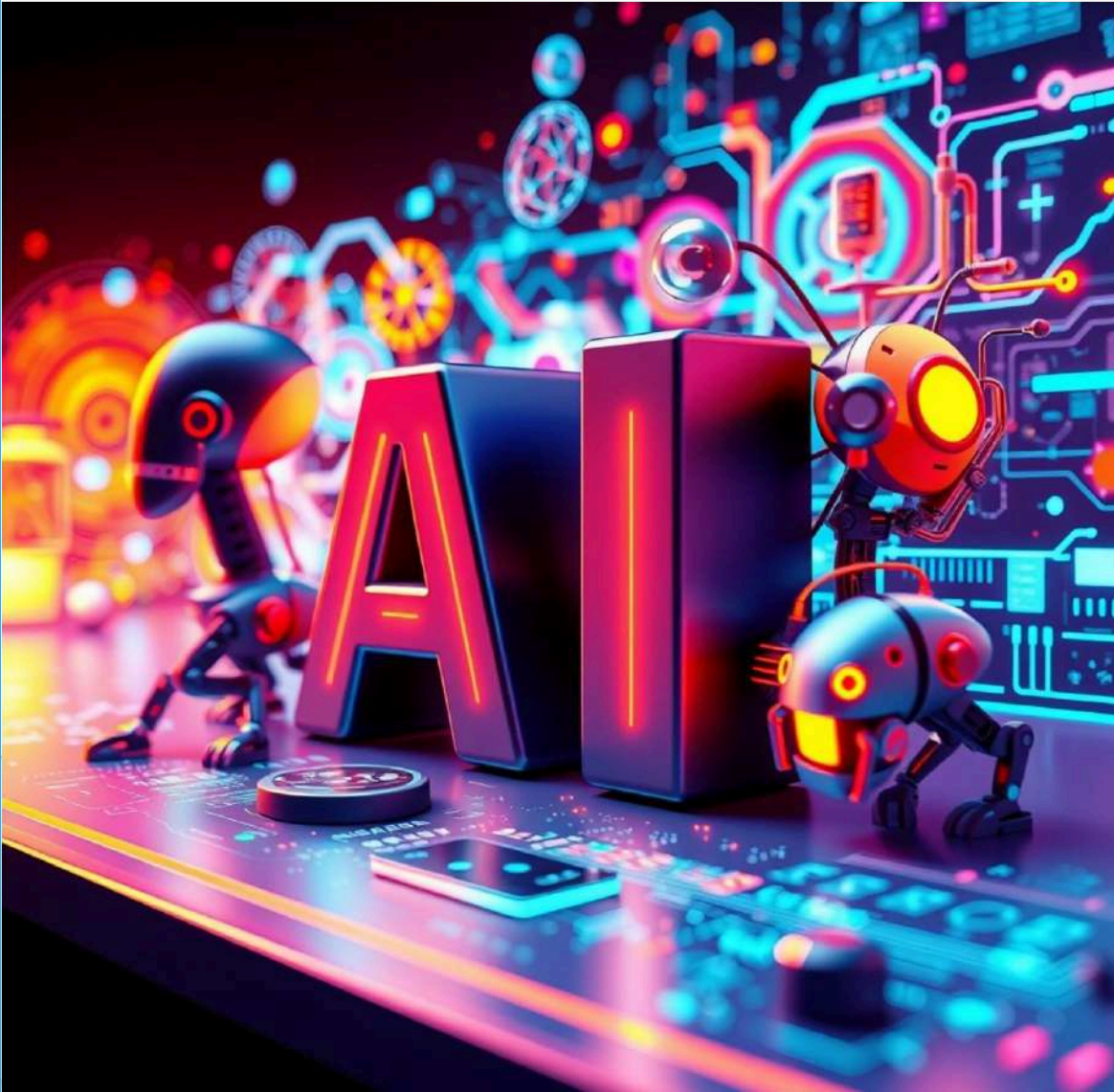
🛠️ Tarea N°2

✦ Diseñar una escena que contenga dos espacios en DescartesJS, expresa tu creatividad.

Ver una posible

✓ [Solución](#)

Sesion 2. 📄 ¡Que aprendiste!



Sesión 3

Diseño de OIA con
DescartesJS e IA

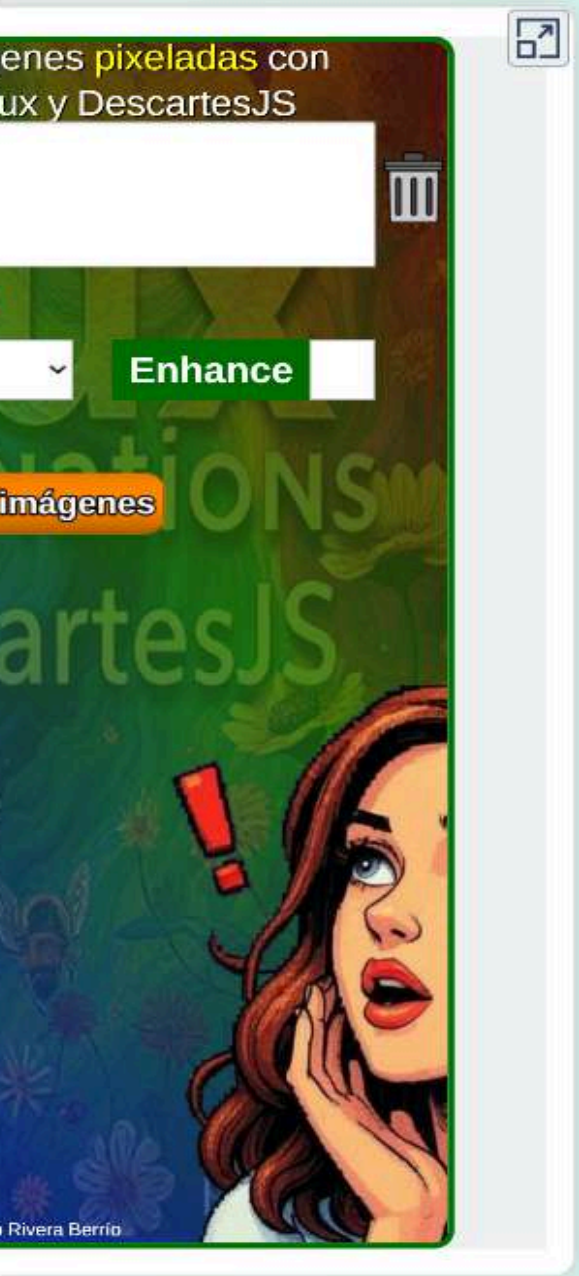
3.1 Los formatos de imagen

¿Qué es una imagen pixelada?

Las imágenes pixeladas, también conocidas como imágenes rasterizadas (mapas de bits), están compuestas por una matriz de píxeles, donde cada píxel representa un color específico. Por ejemplo, si la imagen es de 8 bits, significa que cada píxel puede tener hasta 256 colores, una de 16 bits puede lograr hasta 65,536 colores por píxel. Obviamente, el segundo ejemplo es de mayor calidad y, en consecuencia, de mayor peso. Estas imágenes tienen una resolución fija y pueden perder calidad al ampliarse, lo que resulta en una apariencia borrosa o dentada.

En contraste, las imágenes vectoriales se crean a partir de ecuaciones matemáticas que definen formas geométricas, como líneas y curvas, permitiendo un escalado infinito sin pérdida de calidad.





En informática, las imágenes pixeladas se utilizan comúnmente en fotografías y gráficos detallados, mientras que las vectoriales son ideales para logotipos, iconos y diseños que requieren flexibilidad de tamaño sin distorsión.

En la escena interactiva, escribe gato azul y luego clic en "Generar imágenes" y obtendrás una imagen retro con pixeles muy definidos.

Las imágenes pixeladas suelen utilizar formatos como JPEG, PNG, BMP, GIF, TIF y WebP, cada uno con características específicas.

- **JPEG (o JPG)**, es ideal para fotografías debido a su compresión, que reduce el tamaño del archivo sin gran pérdida de calidad.
- **PNG**, es preferido para gráficos con transparencia, ya que admite compresión sin pérdida.

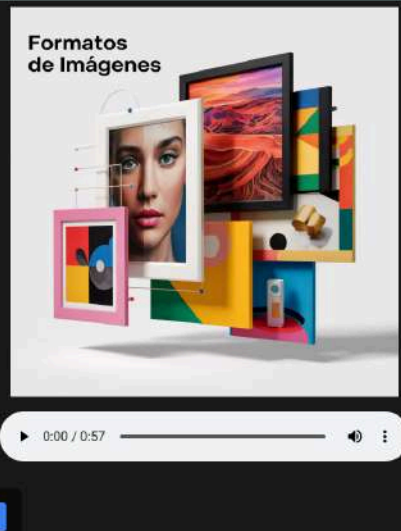
- **BMP**, es un formato sin compresión, lo que resulta en archivos de gran tamaño y alta calidad.
- **GIF**, se emplea para imágenes animadas y gráficos con paletas de colores limitadas y soporta transparencias simples.
- **TIF**, presenta alta calidad para impresión y archivos escaneados (existe con y sin compresión).
- **WebP**, desarrollado por Google, ofrece una compresión más eficiente que JPEG y PNG, permitiendo archivos más ligeros sin perder calidad, además de admitir transparencia y animaciones.



Haz clic sobre la siguiente imagen, para que conozcas otros formatos de imagen.

Formatos de imágenes

Existen diversos formatos de imágenes, cada uno con características específicas que los hacen adecuados para diferentes usos. Los formatos más comunes incluyen JPEG, que es ideal para fotografías debido a su compresión con pérdida; PNG, que admite transparencias y es adecuado para gráficos y logotipos; GIF, que permite animaciones cortas y soporta una paleta limitada de colores; y TIFF, que es utilizado en impresión y edición de alta calidad debido a su compresión sin pérdida. Otros formatos incluyen BMP, que es un formato de imagen sin compresión y PSD, que es el formato nativo de Adobe Photoshop, utilizado para el trabajo creativo. La elección del formato dependerá de la necesidad particular de calidad, tamaño de archivo y uso específico en medios digitales o impresos.



Anterior

1/8

Siguiente

3.2 Imágenes en DescartesJS

Imágenes en espacios

Imagen como fondo de un espacios. DescartesJS maneja sin problema imágenes png, webp, svg, gif y jpg. Por ejemplo, si un archivo portada.png está en una carpeta /imagenes que está en el mismo nivel que el archivo de DescartesJS, entonces se introduciría images/portada.png en el campo de texto para desplegar dicha portada como fondo.

Imagen como gráfico. Este gráfico consiste en una imagen tipo jpg, png, gif, svg (imagen tipo vectorial) o webp que puede insertarse en el interactivo. La imagen debe estar en la misma carpeta que el interactivo o en una subcarpeta. Cabe mencionar que las imágenes webp funcionan en todos los navegadores modernos; y en Safari sólo funciona a partir de la versión Big Sur de mac OS.

 **Guía en video.**



Imágenes en controles y región de recorte

En el apartado anterior, se observó que es posible introducir imágenes en los interactivos. A éstas se les puede controlar su tamaño y proporción, su colocación y su rotación. Aunque todos los parámetros en el ejercicio son números, también pueden usarse variables que permiten hacer estos cambios de forma interactiva. No obstante, esto se verá cuando se aborden ejercicios más avanzados. Determinados controles, como los controles gráficos o los controles tipo botón, permiten introducir imágenes. Para ello, se introduce la dirección relativa de la imagen respecto de la escena en el parámetro correspondiente. En el caso de los controles gráficos, el control es sustituido por la imagen introducida.



Guía en video.

En el video, se indica cómo usar el parámetro región de recorte con los siguientes datos: la posición de la esquina superior izquierda del rectángulo, la anchura y altura del mismo.



Aplicación: imágenes asociadas a un control gráfico

Con lo visto anteriormente, podemos pensar en muchas aplicaciones que usen imágenes en un objeto interactivo. Una de esas aplicaciones es el diseño de presentadores de diapositivas, la cual es la tarea de esta sesión.

Sin embargo, antes de ir a la actividad, presentamos un video en el cual se explica cómo asociar las imágenes a un control gráfico.



Guía en video.

Una vez hayas visto el video, seguramente empezarás a pensar en posibles aplicaciones como escenas de arrastre para el diseño de puzzles o un ejercicio para arrastrar imágenes a contenedores con descripciones de la imagen y un muy largo etcétera.



3.3 Presentadores de diapositivas

Presentadores de diapositivas con DescartesJS

En este apartado vamos a diseñar, con DescartesJS, un presentador de diapositivas de imágenes.

Observa el video tutorial con mucho cuidado y ve construyendo tu presentador. En este ejercicio aprenderás nuevos conceptos, como: control tipo botón, vectores, el condicional dibujar si, un estilo extra, proporciones de la imagen, y... pues... nada más, pocos conocimientos para una gran escena interactiva.



Presentadores de diapositivas con IA

Otra opción interesante es diseñar nuestro propio presentador de diapositivas con ayuda de la Inteligencia Artificial.

Con el siguiente prompt podemos obtener una presentación como la que se muestra a la derecha:

Prompt

Haz un archivo único HTML5 (HTML5 + CSS + JS) adaptable (responsive), para un presentador de diapositivas sobre ciudades europeas. El presentador debe mostrar una imagen, un título y una descripción de la imagen. También debe tener dos botones: uno para ir a la diapositiva anterior y otro para ir a la siguiente. Cuando cambie la diapositiva, debe verse un efecto de transición suave (por ejemplo, que el contenido se deslice o se desvanezca). El contenido (títulos, imágenes y textos) debe poder cambiarse fácilmente. La página debe verse bien tanto en computadoras como en tablets y teléfonos, de forma que sea adaptable a sus pantallas. Genera las imágenes con la función <https://image.pollinations.ai/prompt/{mensaje}>, donde mensaje es el título de la diapositiva.



Una vez obtenido el presentador, podemos modificarlo, cambiando sus contenidos, imágenes, añadiendo o quitando diapositivas, a través de la intervención sencilla de su código, tal y como se indica en el [video](#).

¿Qué es un Prompt?

Un prompt es una instrucción que le das a la IA. La calidad de este prompt define la calidad del resultado. Es la diferencia entre una herramienta útil y una confusa.

Diferencias entre un buen Prompt

Un buen prompt impacta el 75% de la respuesta. Es por esto que la efectividad es la clave para desbloquear el verdadero potencial de la inteligencia artificial.

No muy efectivo:

"Ideas para una clase de historia"

Efectivo:

"Genera 5 ideas de actividades interactivas para una clase sobre la Revolución Industrial, para estudiantes de 15 años, que fomente el análisis de fuentes primarias"

El impacto de un buen Prompt.

Contexto Información de fondo que sitúa la solicitud.

Instrucciones Órdenes claras y directas sobre qué hacer.

Formato Especificación de cómo estructurar la salida

Rol (Personalidad) Asignar una identidad o perspectiva a la IA

Audiencia A quién va dirigida la respuesta

Ejemplos y Restricciones Muestras del resultado esperado y limitaciones o condiciones para la respuesta

🎓 Como docentes, sabemos que estamos frente a una nueva era en la educación, donde la inteligencia artificial se convierte en una aliada estratégica para la enseñanza.

La Era Dorada de los Juegos Retro

Breve introducción sobre qué son los juegos retro, su importancia en la historia de los videojuegos, y un vistazo a las décadas más influyentes (70s, 80s, 90s).



En el [capítulo 2 del libro de plantillas con IA](#), se pueden encontrar diferentes modelos de plantillas de presentadores de diapositivas generadas con ayuda de IA.

Edita código

Copiar al portapapeles

Ejecutar »

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-
width, initial-scale=1.0">
  <title>Presentación de Juegos Retro</title>
  <style>
    body, html {
      margin: 0;
      padding: 0;
      font-family: Arial, sans-serif;
      height: 100%;
      overflow: hidden;
    }
    .presentation {
      position: relative;
      width: 100%;
      height: 100%;
      background-color: #1a1a1a;
      color: white;
    }
    .slide {
      position: absolute;
      top: 0;
      left: 0;
      width: 100%;
      height: 100%;
```



La Era Dorada de los Juegos Retro

Breve introducción sobre qué son los juegos retro, su importancia en la historia de los videojuegos, y un vistazo a las décadas más influyentes (70s, 80s, 90s).



Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)

Proyecto 1 - Generador de imágenes con IA (Parte 2)

Continuando con el primer proyecto con IA, hemos usado nuevos parámetros para la generación de imágenes con Pollinations: dimensiones y estilos de la imagen, que para DescartesJS obligó a usar nuevos controles (campo de texto y menú) y un vector.

Guía en video. (Parte 2)

Observa el video y continúa construyendo el proyecto en tu computador, con tu propio diseño y distribución de espacios.



3.4 🎬 Video - Sesión 3ª

📌 *Diseño de objetos interactivos con DescartesJS e IA*



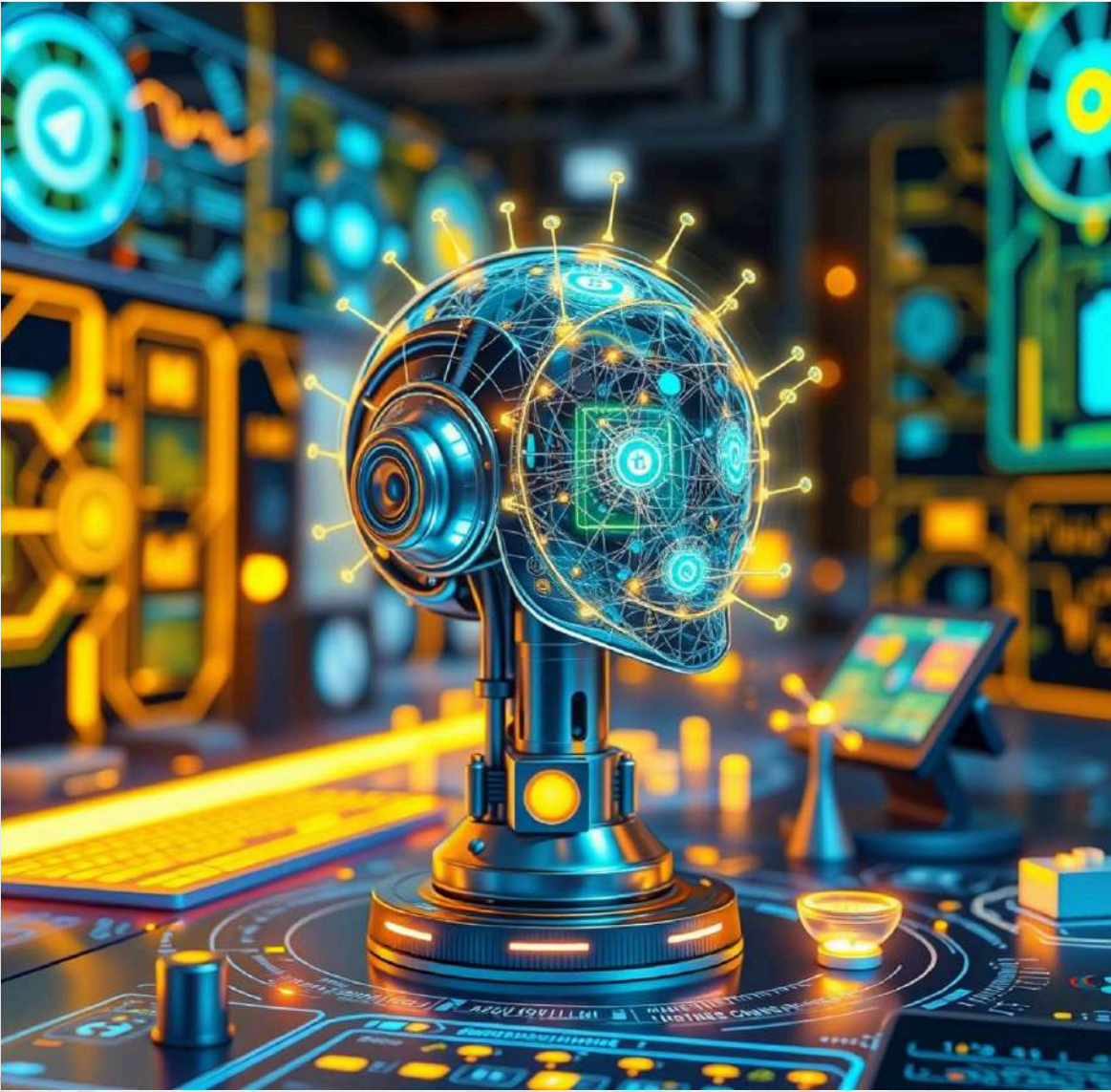
🧰 Tarea N°3

✦ Diseñar un presentador de diapositivas.
¡Usa inteligencia humana!.

Ver una posible

✓ [Solución](#)

Sesion 3. 📄 ¡Que aprendiste!



Sesión 4

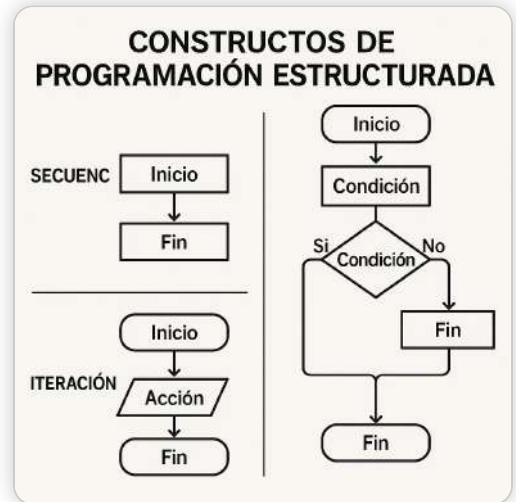
Teorema del Programa Estructurado

4.1 Nivel 1 - Introducción

Definición Básica del Teorema

Descripción:

El Teorema del Programa Estructurado, también conocido como el Teorema de Böhm-Jacopini, establece que cualquier programa computacional puede ser escrito utilizando solo tres estructuras de control: secuencia (ejecución de una instrucción tras otra), selección (instrucciones if-then-else) e iteración (bucles while). Estas tres estructuras son suficientes para expresar cualquier función computable.



Importancia:

Este teorema es fundamental en la ciencia de la computación porque demuestra que la programación estructurada es completa, es decir, no se necesitan instrucciones de salto incondicional (GOTO) para implementar cualquier algoritmo. Esto llevó a mejores prácticas de programación, código más legible y mantenible, y sentó las bases para los lenguajes de programación modernos.

Aplicación en Lenguajes de Programación

Descripción:

Los lenguajes de programación modernos como Java, Python, C++, y JavaScript implementan directamente las tres estructuras básicas del teorema. Utilizan palabras clave como 'if', 'else', 'while', 'for', y bloques de código secuenciales para permitir a los programadores escribir código estructurado sin necesidad de instrucciones GOTO.



Importancia:

La adopción generalizada de estas estructuras ha llevado a una mayor consistencia entre lenguajes de programación, facilitando el aprendizaje y la transición entre ellos.

También ha permitido el desarrollo de herramientas de análisis estático y técnicas de verificación formal que mejoran la calidad y fiabilidad del software.



Impacto en el Desarrollo de Software

El teorema del programa estructurado ha influido profundamente en las metodologías de desarrollo de software, promoviendo enfoques modulares y jerárquicos. Los equipos de desarrollo utilizan estas estructuras básicas para crear sistemas complejos a través de la composición de componentes más simples y bien definidos.



Importancia:

Este enfoque estructurado ha mejorado la colaboración en equipos de desarrollo, facilitado la reutilización de código, y reducido la complejidad ciclomática de los programas. También ha permitido el desarrollo de patrones de diseño y arquitecturas de software que son fundamentales en la industria actual.

Impacto en la forma de programar:

Programación modular: El teorema fomenta la división de programas en módulos más pequeños y manejables, lo que simplifica el desarrollo y el mantenimiento. Mejora de la productividad: Al facilitar la lectura y comprensión del código, la programación estructurada puede aumentar la productividad de los desarrolladores.

Reducción de errores: La eliminación de estructuras de control complejas y la promoción de una estructura clara y secuencial ayudan a reducir la probabilidad de errores en el código.

4.2 Nivel 2 - Profundización

Historia y Origen del Teorema

Descripción:

El teorema fue formulado y demostrado por los matemáticos italianos Corrado Böhm y Giuseppe Jacopini en su artículo de 1966 titulado "Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules". Su trabajo surgió en un momento en que la programación con instrucciones GOTO era común, pero estaba llevando a código difícil de entender y mantener, conocido como "código espagueti".



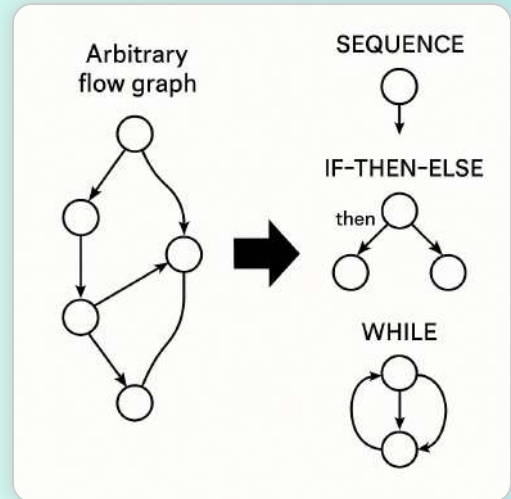
Importancia:

Este trabajo académico proporcionó la base teórica para la revolución de la programación estructurada que siguió en los años 70. Edsger W. Dijkstra, en su famosa carta "Go To Statement Considered Harmful" (1968), citó implícitamente este teorema para argumentar contra el uso de GOTO, iniciando un cambio fundamental en las prácticas de programación.

Demostración Matemática

Descripción:

La demostración del teorema implica mostrar que cualquier diagrama de flujo (que representa un programa) puede ser transformado en un diagrama equivalente que utiliza solo las tres estructuras básicas. Esto se logra mediante la introducción de variables de estado adicionales y la reestructuración del flujo de control, un proceso conocido como "eliminación estructurada de GOTO".



Importancia:

La demostración no solo es teóricamente significativa, sino que también proporciona un método constructivo para transformar programas no estructurados en programas estructurados. Sin embargo, esta transformación puede introducir complejidad adicional, lo que llevó a debates sobre si la programación estructurada estricta siempre resulta en código más claro.

Limitaciones Prácticas

Descripción:

Aunque el teorema demuestra que las tres estructuras básicas son suficientes en teoría, en la práctica hay situaciones donde una adherencia estricta puede llevar a código más complejo o menos eficiente. Ejemplos incluyen el manejo de múltiples condiciones de salida en bucles, la implementación de máquinas de estado finito complejas, o el manejo de excepciones.



Importancia:

Estas limitaciones llevaron a la introducción de construcciones adicionales en lenguajes modernos, como instrucciones break, continue, return en medio de funciones, y manejo de excepciones. Estos pueden verse como extensiones controladas que mantienen el espíritu de la programación estructurada mientras abordan sus limitaciones prácticas.



4.3 Estructuras secuenciales y selectivas

4.3.1 Estructuras secuenciales



Algoritmos



Descripción:

El término "algoritmo" se refiere a un conjunto de instrucciones o reglas definidas paso a paso que se siguen para realizar una tarea o resolver un problema específico. Los algoritmos son fundamentales en la informática, matemáticas y en muchas otras áreas.

Características principales:

Finitud (el algoritmo debe terminar después de un número finito de pasos), Claridad (cada paso debe estar claramente definido y ser comprensible), Entrada (opcionalmente, puede recibir datos de entrada), Salida (produce al menos un resultado o solución y Efectividad (cada paso debe ser realizable).

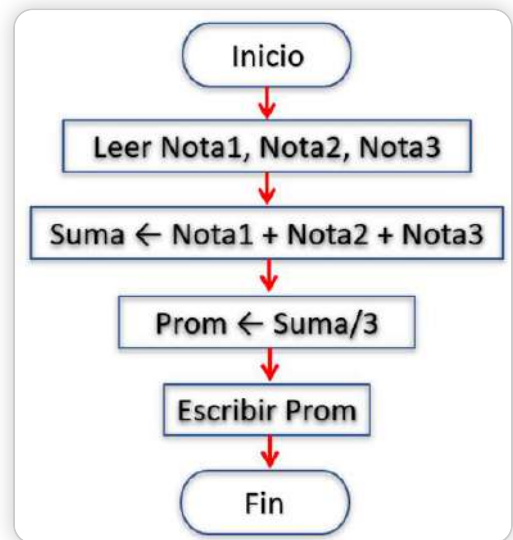
Importancia:

Los algoritmos son importantes porque permiten a los computadores resolver problemas de manera eficiente y estructurada, optimizando procesos y facilitando la toma de decisiones. En la vida cotidiana, los algoritmos son clave para la automatización de tareas, la personalización de experiencias y la detección de patrones, como en redes sociales y sistemas de recomendación.



Asignaciones

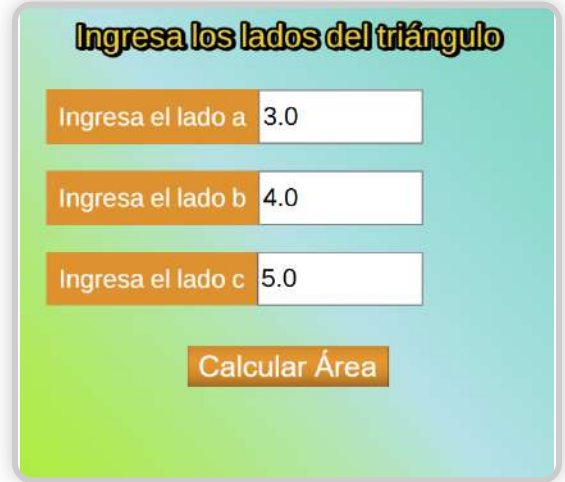
Una asignación en un algoritmo se refiere a una instrucción o conjunto de instrucciones donde se realiza la asignación de valores a variables. La asignación es una operación fundamental en programación y algoritmos que consiste en almacenar un valor en una variable para su posterior uso. En pseudocódigo o diagramas de flujo, los bloques de asignación suelen representarse con un rectángulo donde se indica la operación de asignación.



Un ejemplo sencillo es leer los lados de un triángulo, luego realizar el cálculo del área y, finalmente, mostrar el resultado, tal como se muestra en el objeto interactivo del siguiente apartado.

Ejemplos

En la ventana derecha, presentamos un ejemplo de un objeto interactivo que usa **lectura** de datos, **asignación** de esos datos a las variables **a**, **b**, y **c**, **cálculo** del área del triángulo e **impresión** del resultado; es decir, cuatro bloques de secuencia.



La codificación en JavaScript, solo para leer el dato del lado A, sería de esta forma:

```
<input type="text" id="inputA" value="Hola">
<script>
    let A = document.getElementById("inputA").value;
    console.log(A);
</script>
```



¡Se complicó el asunto!... ¡Claro que no!, con el editor de DescartesJS, la codificación es muy sencilla. En el siguiente video, puedes observar tres ejemplos.

La estructura secuencial

La **estructura secuencial** en programación se refiere a un tipo de algoritmo o programa donde las instrucciones se ejecutan una tras otra, en el orden en que están escritas, sin saltos ni repeticiones. Es la forma más básica de controlar el flujo de un programa, donde cada paso sigue al anterior de manera lineal.

Guía en video.

En el siguiente video se observa como se diseña la escena del ejemplo de la pagina anterior, el cálculo del área de un triángulo.



4.3.2 Estructuras selectivas



Estructura IF simple

Descripción:

La estructura **IF simple** es la forma más básica de control de flujo condicional en programación. Evalúa una condición y ejecuta un bloque de código solo si esa condición es verdadera. Si la condición es falsa, el programa simplemente salta ese bloque y continúa con la siguiente instrucción.

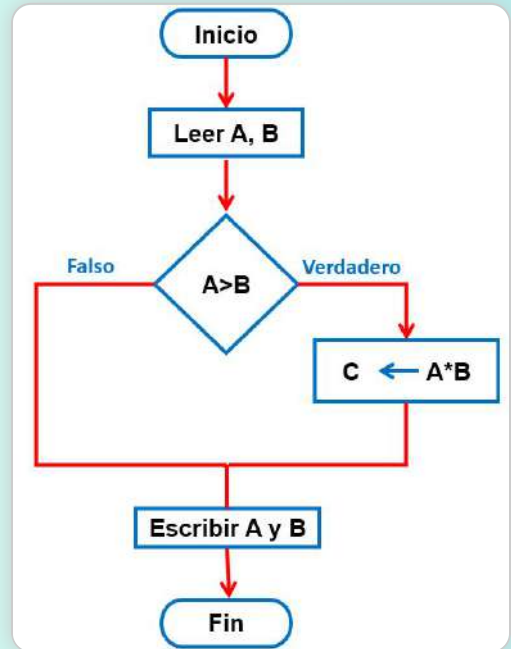
Importancia:

Es fundamental para implementar la lógica de decisión en cualquier programa. Permite que el software responda de manera diferente según las circunstancias, lo que es esencial para crear aplicaciones interactivas y dinámicas.

Ejemplo:

En la imagen se leen los datos de las variables A y B, si y solo si A es mayor que B, se asigna el producto $A*B$ a la variable C. En el editor DescartesJS, sería:

$C = (A > B) ? A * B$

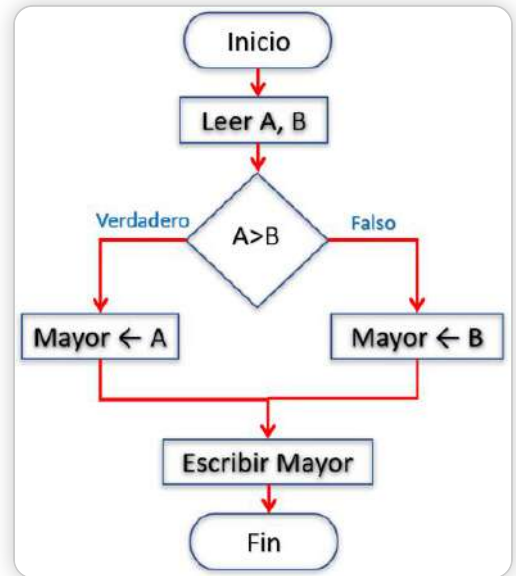




Estructura IF-ELSE y operador ternario de JS

Descripción:

La estructura **IF-ELSE** extiende el IF simple añadiendo un bloque alternativo de código que se ejecuta cuando la condición es falsa. Esto crea una bifurcación en el flujo del programa: si la condición es verdadera, se ejecuta un bloque; si es falsa, se ejecuta el otro.



Importancia:

Permite manejar dos situaciones mutuamente excluyentes de manera clara y eficiente. Es esencial para implementar lógica binaria donde siempre debe ejecutarse exactamente una de dos opciones posibles.

Operador ternario de JavaScript:

En el editor DescartesJS, el condicional de la imagen, se codificaría así: **Mayor = (A>B)? A:B**, expresión conocida como **operador ternario**, cuya sintaxis es:

condición ? valor_si_verdadero : valor_si_falso



Operadores lógicos y de comparación

Operadores de comparación:

Los operadores de comparación son símbolos que permiten comparar dos valores y devolver un resultado booleano (verdadero o falso). Incluyen igual a (==), idéntico a (===), mayor que (>), menor que (<), mayor o igual que (>=), menor o igual que (<=), y no igual a (!=).

Operadores lógicos:

Los operadores lógicos permiten combinar múltiples condiciones en una sola expresión. Los principales son AND (&), OR (||) y NOT (!). AND devuelve verdadero solo si ambas condiciones son verdaderas; OR devuelve verdadero si al menos una condición es verdadera; NOT invierte el valor de una condición.

Ejercicio:

En los operadores anteriores, hemos puesto, entre paréntesis, los operadores usados en JavaScript y, en consecuencia, en el editor DescartesJS.

Al lado derecho presentamos un objeto interactivo, como ejercicio cambia los valores de A y B y observa cuáles operadores son verdaderos (ver [aquí](#) en pantalla completa).

El último operador (fondo verde), que llamamos c7, se codificó así:

```
c7=(A=3)&!(B=3)?1:0
```

Operadores de comparación

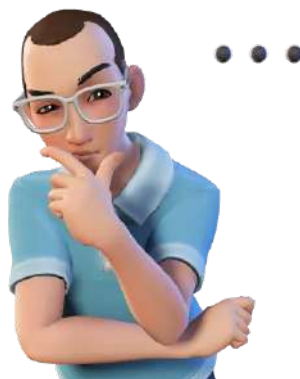
 ☒ ☒

Operadores lógicos

Ejemplo en DescartesJS

Hallar el número mayor.

En la [figura](#) de [la estructura IF-ELSE](#), presentamos un algoritmo en forma de diagrama de flujo, para hallar el número mayor entre dos números, usando un condicional o estructura selectiva.



Presentamos un objeto interactivo, diseñado con DescartesJS, codificado según el algoritmo, permitiendo hallar el número mayor entre dos números. (ver [aquí](#) en pantalla completa).

Guía en video.

A continuación, puedes observar un video que explica cómo se usó el operador ternario de JavaScript, el cual también usaremos para terminar nuestro **Proyecto 1: "Generador de imágenes con IA"**.

Presta atención, al inicio del video, pues te explicaremos la utilidad de la opción **mostrar región exterior**, del selector escena.

Hallar el número mayor

Ingresar el primer número

0

Ingresar el segundo número

0

Hallar mayor



Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)

Proyecto 1 - Generador de imágenes con IA (Parte 3)

Concluimos el primer proyecto de aula con IA usando nuevos parámetros para la generación de imágenes con Pollinations como semilla y el parámetro nologo, para ocultar el logo de Pollinations. También, hemos incorporado nuevos elementos de DescartesJS, como el control tipo texto, la función genera() y la descargas de imágenes abriendo una URL en una pestaña nueva con la imagen.

Guía en video. (Parte 3)

Observa el video y termina de diseñar el proyecto en tu computador.



4.4 🎬 Video - Sesión 4ª

📌 Diseño de objetos interactivos con DescartesJS e IA



📦 Tarea N°4

✦ Resuelve los los siguientes problemas:

Problema 1. Usa una IA (herramienta 46, por ejemplo), para calcular: "En un almacén se tiene la siguiente promoción: todo artículo con un precio superior a \$25,000 tendrá un descuento del 20%, los demás tendrán el 10%. Determinar el precio final que debe pagar una persona por comprar un artículo y el valor del descuento."

Ver una posible

✓ [Solución](#)

Problema 2. Crea un programa en DescartesJS que reciba una calificación numérica entre 0 y 100 e imprima el nivel correspondiente:

- 90 a 100: Excelente
- 80 a 89: Notable
- 70 a 79: Aprobado
- Menos de 70: Reprobado



Ver una posible

 [Solución](#)

Sesion 4.  [¡Que aprendiste!](#)



Sesión 5

Estructuras iterativas

5.1 Nivel 1 - Introducción

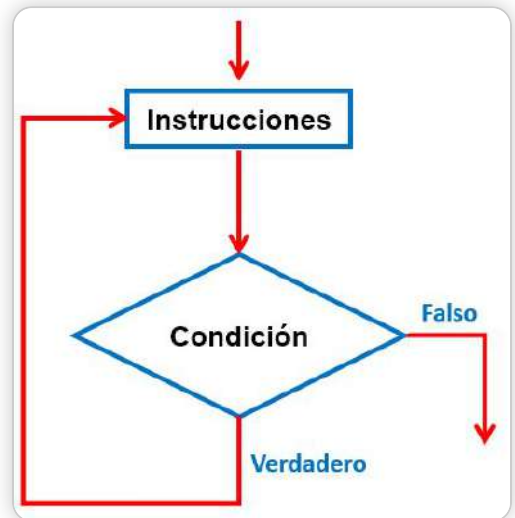
Definición estructura iterativa

Descripción:

Una estructura iterativa o de repetición es aquella que permite que un proceso o conjunto de acciones se realice de manera cíclica (loop o bucle). Este proceso repetitivo se realiza mientras una condición lógica sea verdadera. Existen cuatro formas comunes de este tipo de estructuras: "Hacer mientras" (Do-While), "Mientras que", "Repite hasta que" y "Desde, hasta" (For-next). En este apartado sólo trabajaremos con la primera forma, pues es la que usa DescartesJS. En la figura de la izquierda se presenta la forma Do-While en un diagrama de flujo.

Importancia:

La estructura iterativa (o bucle) es fundamental en programación porque permite repetir un conjunto de instrucciones múltiples veces, lo que proporciona eficiencia, claridad y control sobre la ejecución de un programa. Un aspecto a destacar es la automatización de tareas repetitivas, pues en lugar de escribir el mismo código varias veces, una estructura iterativa ejecuta una acción repetidamente.



Ejemplo de uso de una estructura iterativa

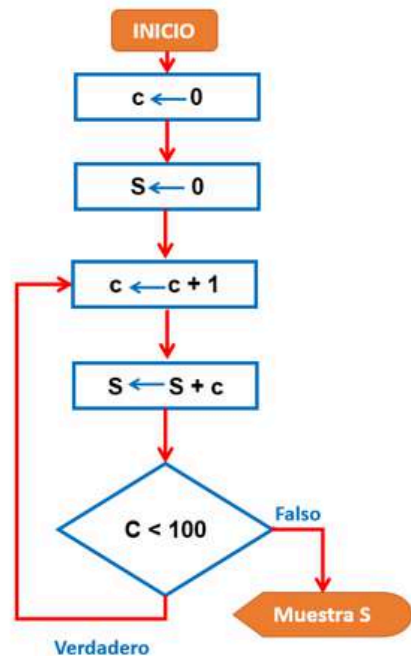


Calcular la suma de los 100 primeros números naturales

Algoritmo en lenguaje natural:

1. Inicio.
2. Crear una variable llamada contador (c) y asignarle el valor 0. Esta variable representará el número actual que se va a sumar.
3. Crear una variable llamada suma (S) y asignarle el valor 0. Esta variable almacenará la suma total.
4. **Hacer** lo siguiente (bloqueado):
Aumentar el valor del contador c en 1 (incrementarlo en 1).
Sumar el valor del contador c a la variable suma S .
5. **Mientras** el valor del contador c sea menor a 100 ($c < 100$), repetir el paso anterior.
6. Cuando el contador es igual 100, salir del bucle.
7. Mostrar el valor final de la variable suma S .

```
1. Inicio
2.  $c \leftarrow 0$ 
3.  $S \leftarrow 0$ 
4. Hacer
     $c \leftarrow c + 1$ 
     $S \leftarrow S + c$ 
    Mientras  $c < 100$ 
Fin Hacer
5. Escribir  $S$ 
6. Fin
```



En la [figura](#), mostramos el algoritmo como pseudo-código y como diagrama de flujo.

Una buena práctica es hacer una [prueba de escritorio](#); por ejemplo, toma papel y lápiz y sigue el algoritmo con los 5 primeros números naturales, la suma te debe dar 15.



Guía en video.

En el siguiente video, puedes observar lo sencillo que es hacerlo con DescartesJS:



5.2 Nivel 2 - Estructuras iterativas y vectores

Vectores

Descripción:

En ocasiones se desea utilizar una gran cantidad de variables que guardarán un tipo similar de información.

En estos casos, puede resultar engorroso crear muchas variables distintas, y conviene en su lugar definir una especie de “mueble con muchos cajones”. Aunque el nombre del mueble es el mismo, cada cajón tiene un número o índice que lo identifica.

Cuando se agrega un vector, sólo es necesario dar su nombre. Por ejemplo, el vector **E**. Una vez agregado, siempre se hará referencia al mismo indicando el número de cajón en cuestión. Por ejemplo, **E[0]** representa el primer cajón, **E[1]** representa el segundo cajón, y así sucesivamente. Como se puede notar, el índice de los cajones empieza siempre en cero. Cada cajón puede guardar un número o cadena de caracteres, como si fuera una variable común y corriente.



Ejemplo de uso de una estructura iterativa con vectores



En un archivo se tienen los datos de la estatura de un grupo de 15 estudiantes.

Elabora un pseudocódigo y ejecútalo en DescartesJS para calcular la estatura promedio.



Solución:

El pseudocódigo que soluciona este problema se muestra en la figura de la derecha. Nótese que se incluye la lectura de las estaturas, lo cual se podría hacer con controles tipo campo de texto; sin embargo, si queremos calcular la estatura promedio de los 4 000 estudiantes de un colegio, la lectura desde controles tipo campo de texto no es una opción conveniente, es aquí donde debemos recurrir a otro tipo de lectura como el uso de un vector que almacene estas estaturas.

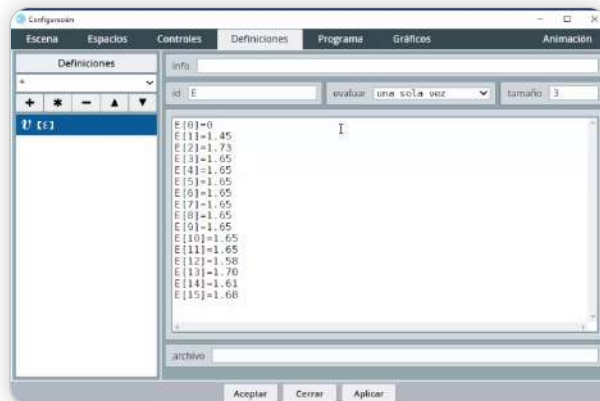
```
1. Inicio
2. i = 0
3. S = 0
4. Hacer
    i = i + 1
    Leer Estatura
    S = S + Estatura
    Mientras i < 15
  Fin Hacer
5. Escribir S/15
6. Fin
```

¿Cómo hacerlo?, observa el siguiente video y practica.

 **Guía en video.**



La imagen siguiente muestre el vector creado con las 15 estaturas de los estudiantes en el menu "Defniciones" del editos DescartesJS:



5.3 Nivel 3 - Lectura de datos externos

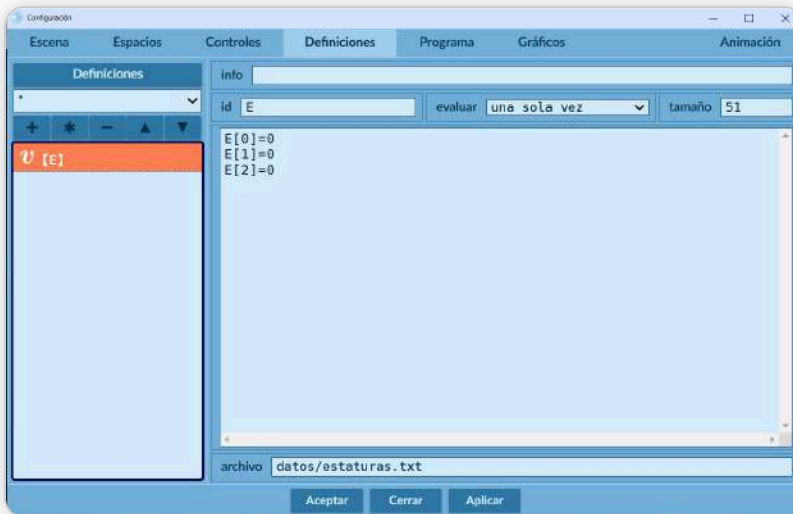
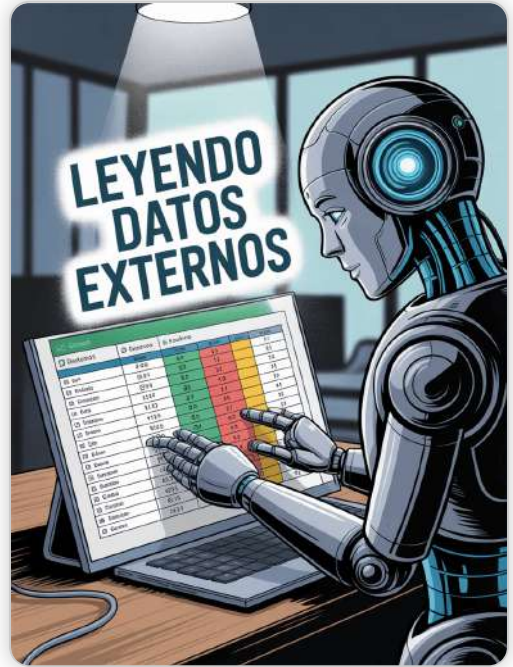
Lectura de datos desde un archivo (fichero) externo

Descripción:

En el video anterior, asignamos 15 estaturas a los elementos de un vector, algo engorroso, pero válido. Pero, ¿qué hacer si tenemos los datos de 50 personas, almacenados en un archivo Excel o en un bloc de notas?, digitarlos no es una opción recomendable.

Lectura de archivos:

DescartesJS permite que podamos leer un archivo en formato txt, tal como se ve en la siguiente imagen.



En el ejemplo del apartado anterior, pese a usar un archivo externo, tuvimos que digitar las 15 edades. En la práctica, esos datos, usualmente, están almacenados en una base de datos, luego que los usuarios han ingresado esa información; por ejemplo, para inscribirte a este curso, ingresaste información a través de un formulario "Google", pero, ¿cómo usarlos en DescartesJS?

Ejercicio práctico con un archivo de Excel

Lectura de archivos:

Una forma sencilla es copiar los datos, pegarlos en un bloc de notas y guardarlos en una archivo txt, para luego leerlos en un vector de DescartesJS; sin embargo, es importante recordar cómo deben leerse datos tipo cadena de caracteres, pues estos deben ir entre comillas simples.

Guía en video.

En el siguiente video, explicamos cómo leer información de 50 registros en una hoja Excel y exportarlos a DescartesJS:



Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)



Proyecto 2 - Chatbot con personalidad (parte 1)

Iniciamos el **2º proyecto de aula**, diseñando un chatbot con la API de pollinations, incluyendo varios modelos de lenguaje. Usaremos, además de la API de Pollinations, elementos de DescartesJS como: espacios, variables de espacio, controles tipo menú, texto y botón, funciones y vectores.



Guía en video. (Proyecto 2 - Parte 1)

En el siguiente video, puedes ver la explicación, paso a paso, de cómo diseñar un chatbot básico, el cual iremos mejorando.



5.4 🎬 Video - Sesión 5ª

📌 Diseño de objetos interactivos con DescartesJS e IA



🧰 Tarea N°5

✦ Resuelve los problemas (descarga este archivo de  [Excel](#)):

Problema 1. Crea un programa en DescartesJS, que calcule:

- ¿Cuántas personas son mayores de 40 años?
- El nombre y edad de la persona de mayor edad
- Cuántas mujeres son mayores de 30 años

Ver una posible

✓ [Solución](#)

Problema 2.

Convierte el archivo a formato PDF con  [iLovePDF](#), sube el archivo al  [Chatbot Investiga](#) y formula 5 preguntas; por ejemplo, ¿cuántas mujeres tienen más de 40 años?. ¿cuántos hombres tienen nombre que empiezan por vocal? o ¿cuántas mujeres mayores de 30 años, miden menos de 1,70?, etc.



Elabora esta tarea con las 5 preguntas y tus opiniones sobre los resultados y el comportamiento del chatbot.

Ver una posible

 [Solución](#)

Sesion 5.  [¡Que aprendiste!](#)



Sesión 6

Los controles del
editor DescartesJS

6.1 Nivel 1 - Introducción

En DescartesJS, los controles son objetos interactivos que permiten al usuario manipular y modificar los elementos de una escena. Estos controles, como pulsadores, barras, menús, y otros, facilitan la interacción del usuario con el contenido dinámico de la escena.

Funcionalidad:

Los controles en DescartesJS son fundamentales para crear escenas interactivas, permitiendo a los usuarios:

- **Modificar parámetros:** Cambiar valores de variables que afectan el comportamiento de la escena.
- **Activar animaciones:** Iniciar o detener animaciones.
- **Ejecutar funciones:** Desencadenar acciones específicas.
- **Navegar entre escenas:** Controlar el flujo de la experiencia interactiva.
- **Ubicación y Uso:** Los controles se pueden encontrar en el editor de DescartesJS, específicamente en el selector de "Controles". Una vez seleccionados, se pueden agregar a la escena y personalizar su apariencia y funcionalidad.

Ubicación y Uso:

Los controles se pueden encontrar en el editor de DescartesJS, específicamente en el selector de "Controles". Una vez seleccionados, se pueden agregar a la escena y personalizar su apariencia y funcionalidad.

El selector controles del editor de configuraciones

Los controles son objetos con los que el usuario, valga la redundancia, controla el objeto interactivo. Éstos permiten que el usuario haga modificaciones a los valores de las variables, lance animaciones, ejecute funciones, así como un gran número de otras acciones. Muchas de estas herramientas de interacción son seguramente familiares para el usuario, como lo son los pulsadores, las barras horizontales, los menús y los botones.

En el selector Controles puedes encontrar 11 tipos de control. Para esta sesión, sólo trabajaremos seis de ellos, algunos ya usados en sesiones anteriores como los controles tipo botón y tipo menú, que incluimos en esta introducción. Los controles de Audio y Vídeo los veremos en la siguiente sesión.

Los controles son fundamentales para la interactividad, permitiendo a los usuarios manipular el contenido dinámico en un objeto de aprendizaje interactivo (OIA), son la interfaz principal a través de la cual los usuarios interactúan con las simulaciones y contenidos interactivos en DescartesJS. Permiten el control del usuario, mejorando el compromiso y la exploración del material.



El control tipo botón

Este tipo de control numérico es el más simple. Consiste en un botón que al ser oprimido hace alguna acción. Muchos de los parámetros del botón son comunes a todos los controles.

Diseño

El diseño del botón se hace poniendo colores interiores y de texto; sin embargo, es posible incluir estilos especiales a través del cuadro de texto estilo extra, entre ellos:

```
overColor=f00| border=3| borderColor=1c42a5|  
borderRadius=15| flat=1
```

Cada estilo debe ir separado con la barra vertical "|". En el objeto interactivo de la página siguiente, destacamos, en los tres primeros botones, algunos estilos usados; por ejemplo, el estilo flat pone o elimina un color de fondo degradado.

Uso

Un uso habitual del control tipo botón es el de hacer cálculos, como el botón "Calcular" del objeto interactivo.

Otro uso interesante es vincular una imagen al botón. Haz clic en "Avanzar", para observar una escena interactiva con dos imágenes vinculadas a botones.



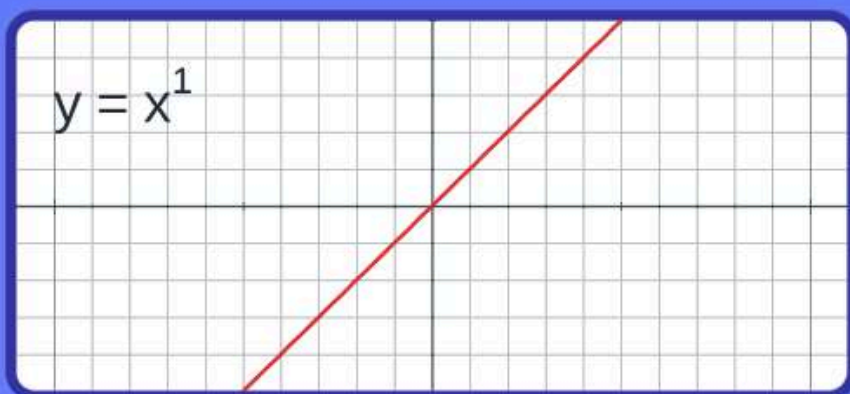
Diseño del control tipo botón

Diseño 1 overColor=aa0 | flat=0

Diseño 2 overColor=aaf | flat=1

Diseño 3 shadowBoxColor=000

Calcular

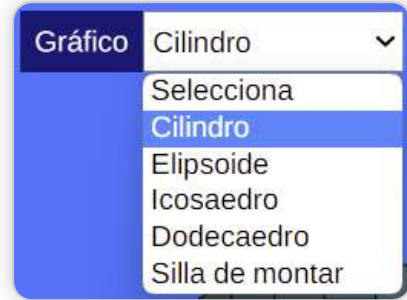


Avanzar

El control tipo menú

Este tipo de control numérico es un menú típico. Resulta principalmente útil para permitirle al usuario tener acceso a alguna opción dentro del interactivo.

Muchos de los parámetros del menú son comunes a todos los controles que hemos visto. En el objeto interactivo de la página siguiente se muestra un ejemplo de un control numérico tipo menú en una escena.



Las opciones del menú se introducen en un campo de texto. Las opciones se separan mediante comas (,). Si se desea que la primera opción esté vacía, se puede colocar un espacio, luego una coma, y luego el resto de las opciones.

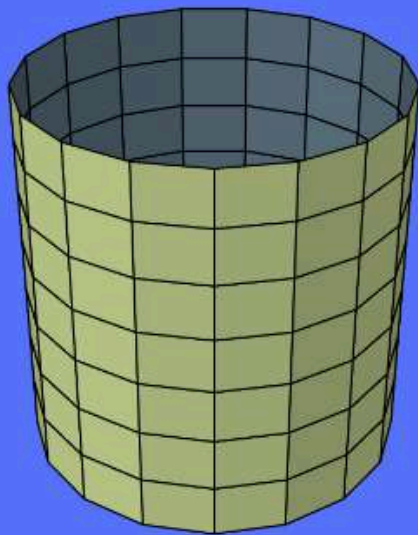
opciones Selecciona,Cilindro,Elipsoide,Icosaedro,Dodecaedro,Silla de montar

Esto se hace cuando se quiere que el menú no tenga una opción particular seleccionada al iniciar el interactivo, sino que aparezca vacío. Las opciones del menú determinan el valor del mismo. La primera opción hace que éste valga 0, la segunda opción hace que valga 1 y así sucesivamente.

En el objeto interactivo "la opción 0" es el mensaje "Selecciona", que para los gráficos de la escena no aplican, pues los objetos gráficos se dibujan con las opciones 1, 2, 3, 4 y 5. Una mejora introducida recientemente, es la posibilidad de capturar el nombre de la opción seleccionada en el menú. Para el ejercicio, el menú lo hemos identificado (id) con la variable m1, de tal forma que el nombre de la opción lo pusimos usando m1.txt.



Gráfico Cilindro ▼



Cilindro

Nu



7



Nv



7



6.2 Nivel 2 - Controles tipo pulsador y tipo barra

Control tipo pulsador

Este tipo de control se asocia con una variable que tiene el mismo nombre del identificador del pulsador. El pulsador permite aumentar y disminuir los valores de la variable, mediante un botón pequeño superior y uno inferior, respectivamente. Adicionalmente, también puede involucrar un campo de texto en el cual se le puede teclear un valor o expresión para modificar su valor.

El estilo del control se puede seleccionar en el menú posición de botones, mediante el cual se determina cómo se disponen los botones de decremento / incremento del pulsador respecto al campo de texto del mismo. Sus opciones son: vertical izquierda, vertical derecha, horizontal izquierda, horizontal derecha y horizontal extremos, tal como se aprecia en el objeto interactivo de la derecha.



Control tipo barra

Este tipo de control numérico consiste en una barra de desplazamiento (o scroll) horizontal o vertical que, al moverla, cambia su valor. Puede servir para moverse de un valor a otro alejado de forma rápida, aunque también permite cambios de valores más finos (desactivando la casilla **fijo**).

En el objeto interactivo, hemos puesto tres controles tipo barra, modificado posición y colores. Hemos asociado una imagen a estos controles, a través de la expresión (p1,p2,p3,p3), de tal forma que el primer control cambia la posición horizontal de la imagen, el segundo control la posición vertical y el tercero la proporción de la imagen.

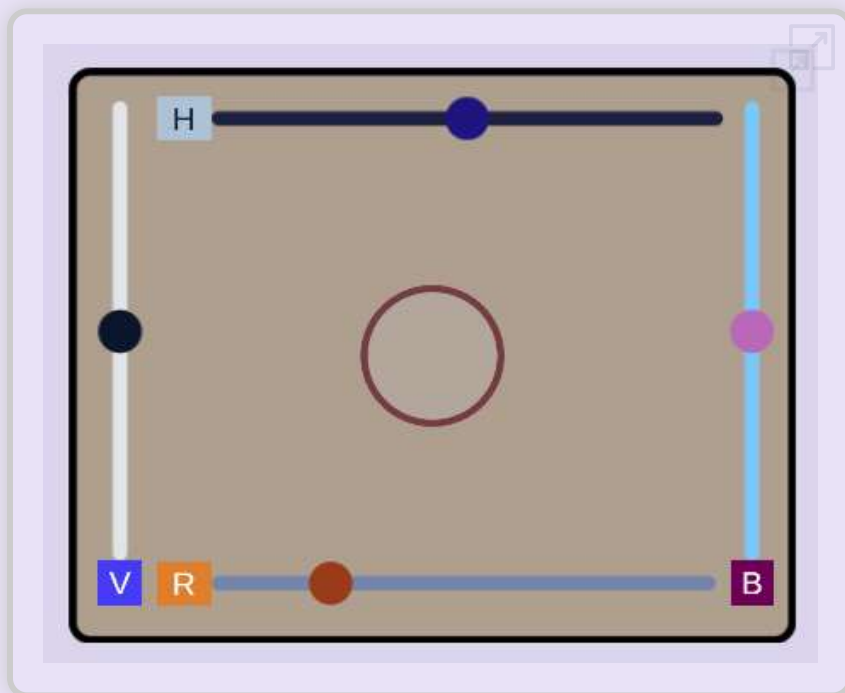


6.3 Nivel 3 - Controles tipo deslizador y casilla de verificación

Controles tipo deslizador

Este control tiene un funcionamiento similar al de la barra de desplazamiento. En este caso un círculo se desliza a través de un eje que puede ser horizontal y vertical. La magnitud de su desplazamiento se define a través de un incremento y el control toma valores acotados entre un mínimo y un máximo. Permite definir el color y tamaño de los diferentes elementos que lo forman.

En el objeto interactivo, hemos colocado cuatro controles tipo deslizador (**p1, p2, p3 y p4**) en distintas posiciones y con colores. Hemos insertado un círculo (elemento arco del menú gráficos) cuya posición, radio y ancho de borde están establecidos por los controles tipo deslizador.

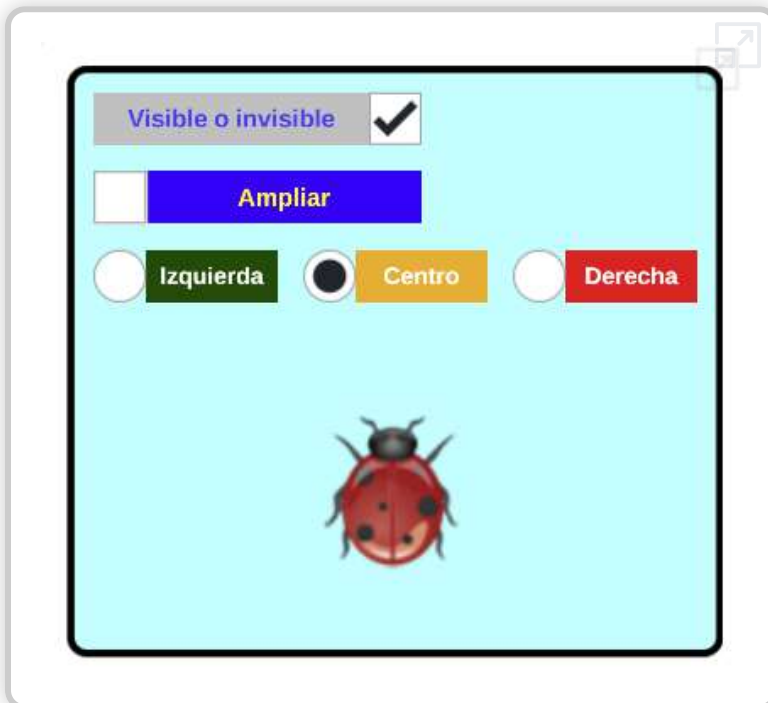


Control tipo casilla de verificación

Este control solo puede tomar dos valores 0 (desmarcado) o 1 (marcado). Su valor puede utilizarse como condicional binario en otras acciones de la escena. Se pueden ajustar el color, tamaño y la posición derecha o izquierda de la casilla de verificación.

Las casillas de verificación se pueden agrupar en un control tipo radio. Para ello, en un parámetro denominado **grupo** se debe poner el mismo nombre a todas las casillas de verificación que pertenezcan a dicho control. Esto hará que solo una de las casillas de verificación del grupo pueda estar marcada a la vez, permitiendo seleccionar un único elemento en este tipo de control.

En el interactivo se han incluido dos casillas de verificación y un control tipo radio y se puede ver como el valor de dichos control se utiliza en realizar diferentes acciones sobre la imagen de la mariquita.



Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)



Proyecto 2 - Chatbot con personalidad (parte 2)

Continuamos el segundo proyecto de aula, incluyendo en nuestro chatbot el texto en modo gráfico, el cual nos permite ver la respuesta en un tamaño de letra más grande y con el color que queramos. Adicionalmente, podemos cambiar la personalidad de nuestro bot conversacional.



Guía en video. (Proyecto 2 - Parte 2)

En el siguiente video, puedes ver la explicación, paso a paso, de cómo implementar estas dos mejoras.



6.4 🎬 Video - Sesión 6ª

📌 Diseño de objetos interactivos con DescartesJS e IA



🧰 Tarea N°6

✚ **DescartesJS.** Usa tu creatividad y diseña una escena interactiva con el editor DescartesJS. La escena debe tener, al menos, dos controles de los vistos en esta sesión.

✚ **IA.** Diseña con IA una actividad que incluya dos tipos de controles (zum y menú, por ejemplo).

Ver una posible

✓ [Solución](#)

Sesion 6. ¡Que aprendiste!



Sesión 7

Audios y videos

7.1 Nivel 1 - Introducción

⚡ Formatos de audios y videos

HTML5 es compatible con una variedad de formatos de audio y video, pero no todos los navegadores admiten todos los formatos.



Formatos de video compatibles con HTML5

MP4 (H.264 + AAC): El más ampliamente soportado por todos los navegadores modernos (Chrome, Firefox, Edge, Safari); WebM (VP8/VP9 + Vorbis/Opus): Optimizado para la web y de código abierto (Compatible con la mayoría de navegadores modernos); Ogg (Theora + Vorbis): Menos común, pero aún admitido por algunos navegadores como Firefox.

Formatos de audio compatibles con HTML5

MP3: Altamente compatible y ampliamente usado; Ogg (Vorbis): Formato libre y abierto, bien soportado; WAV: Alta calidad, pero archivos grandes. Soportado por la mayoría de navegadores.



⚡ Los audios y videos en el editor DescartesJS

Los objetos interactivos de DescartesJS permiten la inclusión de elementos de audio y vídeo. Estos pueden ser introducidos de varias formas:

A través de un espacio HTMLiframe que puede llevar incorporado cualquier objeto soportado por una página web, incluidos los elementos de audio y vídeo.

Mediante un evento que permite reproducir archivos de sonido.

Utilizando controles de audio y vídeo y sus funciones asociadas. En esta sesión nos centraremos en esta opción.



⚡ El control tipo audio en DescartesJS

Este control consiste en un reproductor de audio que puede ser activado en el interactivo. Se puede colocar sólo en el interior de un espacio. Reproduce archivos mp3 y wav.

Configuración

La posición y el tamaño del reproductor se indican mediante el campo expresión, del tipo (x,y,a,b), en el que los dos primeros datos indican la posición y los dos últimos el ancho y el alto del reproductor.

En el campo archivo se debe introducir la ruta del fichero de sonido que se desea reproducir.

Además, al igual que otros controles dispone de los campos: identificador (id), espacio y dibujar-si.

Funciones asociadas

Al ser invocadas las siguientes funciones: `id.play()`, `id.stop()`, `id.pause()` e `id.currentTime(t)`, inician, detienen, paran o colocan el reproductor un tiempo determinado respectivamente. (ver [aquí](#) en pantalla completa).



⚡ El control tipo vídeo en DescartesJS

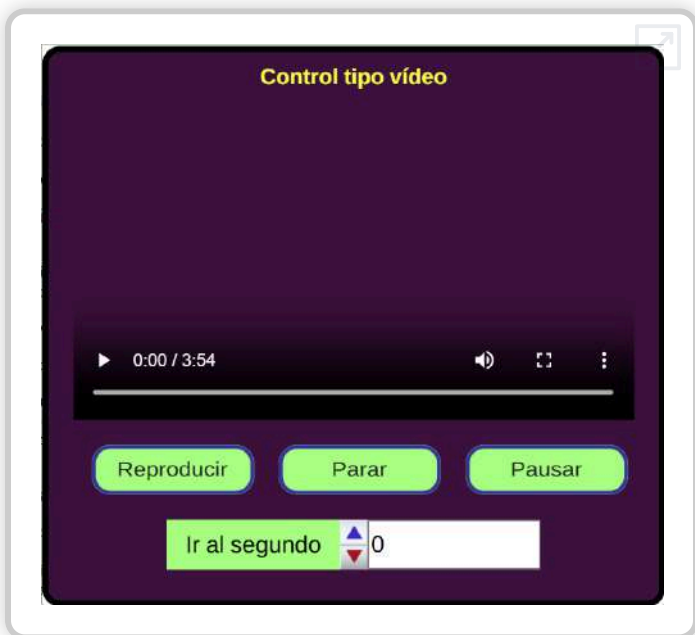
Este control consiste en un reproductor de video que puede ser activado en el interactivo. Su funcionamiento es muy similar al control de audio. Se utiliza para reproducir archivos con formato mp4, WebM y ogv.

Configuración

Es la misma que la del tipo audio. Se debe tener en cuenta que las dimensiones deben ser las adecuadas para introducir un vídeo. Tanto en este control como en el de tipo audio se puede incorporar vídeos con una ruta relativa al interactivo o vídeos a través de una dirección de Internet.

Funciones asociadas

Son las mismas que las de tipo audio. Estas funciones pueden ser llamadas desde cualquier control, programa o evento.



7.2 Nivel 2 - Diseño de vídeos interactivos con el editor DescartesJS

⚡ Obtención o generación de videos en local

Reproducir videos en local mediante HTML5 es una práctica común cuando se desea que un sitio web funcione sin depender de una conexión a Internet para acceder al contenido multimedia (como YouTube o Vimeo).

✓ **Ventajas:** Funciona sin conexión a Internet, control total sobre el archivo y mejor privacidad (no se conecta a servicios externos).

⚠ **Desventajas:** Mayor peso del archivo HTML general si se incluyen videos muy pesados.

🎥 Guía en video.

Obtener o generar un video para desarrollar la actividad de este apartado; para ello, observa el video.



⚡ Actividades interactivas a incluir en el video

La clave del video interactivo es que transforma al espectador en un participante activo, mejorando el compromiso, la retención de información y la personalización del contenido. Por ejemplo, en un video interactivo educativo, el usuario puede responder preguntas en ciertos momentos.

En esta segunda parte, el propósito es definir las actividades interactivas que vamos a usar en el video interactivo; para ello, sugerimos usar las **plantillas** del Proyecto Descartes, tal como se indica en el video.

🎬 Guía en video.



(Haz clic [aquí](#) , para descargar las plantillas, mencionadas en el video).

⚡ Diseño del video interactivo

Es importante recordar que el objetivo del video interactivo es permitir al usuario interactuar con la escena, en la cual el objeto principal es el video.



En el siguiente video, puedes observar, con detalle, las funciones, variables, espacios, controles y demás elementos del editor DescartesJS, usados para el diseño final del video interactivo. A continuación, presentamos el video interactivo obtenido, al cual le hemos puesto un texto con el tiempo transcurrido y la línea de tiempo con marcas indicando dónde se presentarán las actividades.

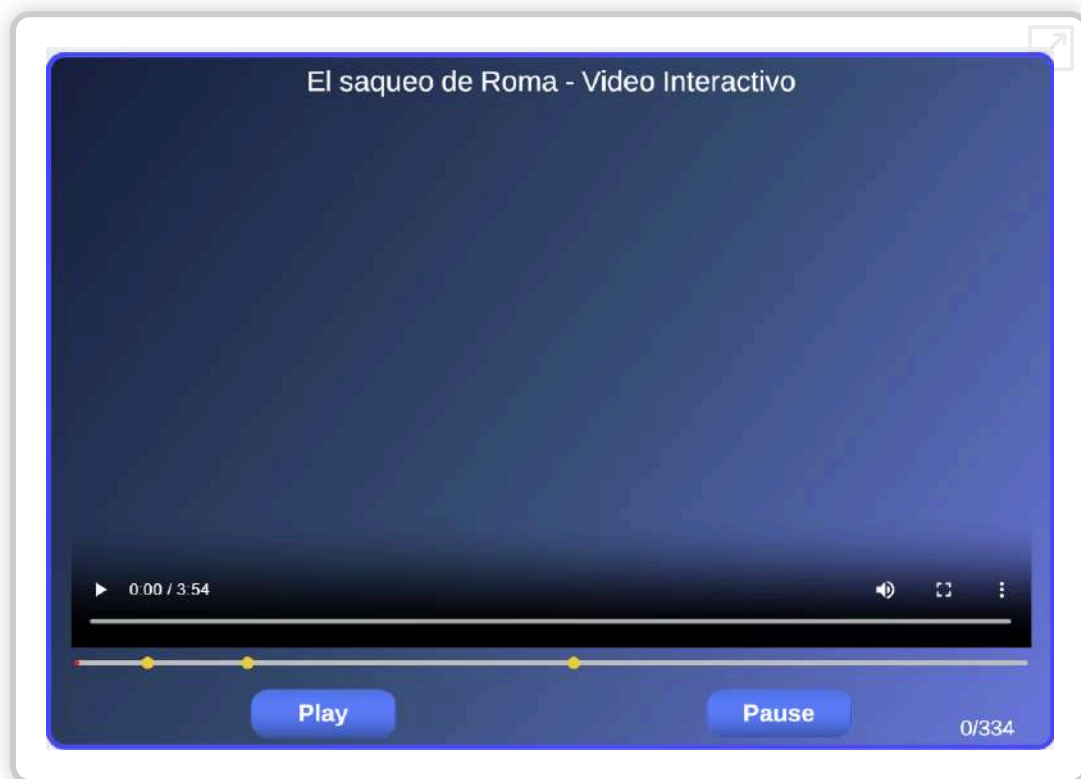
🎬 Guía en video.



La **interacción en un video**, se realiza a través de preguntas o de la selección de opciones presentadas como botones o puntos gráficos dibujados sobre el vídeo.

Esta interacción se logra si sólo se habilitan los controles de vídeo antes mencionados, que sólo permitan al usuario interferir en la reproducción de acuerdo a una intencionalidad didáctica.

La comunicación bidireccional escena - HTML posibilita una gran variedad de alternativas o modelos de vídeos interactivos.



7.3 Nivel 3 Diseño de vídeos interactivos con IA

⚡ Plantillas para videos interactivos

Un libro guía para diseñar video interactivos

🔗 "[Plantillas para libros con inteligencia artificial](#)"

En un mundo donde la tecnología avanza a pasos agigantados, la inteligencia artificial (IA) se ha convertido en una herramienta esencial para la educación y la creación de contenidos, entre ellos los videos interactivos. En el libro, se presentan plantillas sencillas de utilizar, las cuales nos permiten generar videos interactivos. Obviamente, es necesario que tengas definido el video, bien sea de YouTube o un video en local, además de las preguntas a incluir en el video.



A continuación, presentamos dos modelos para diseñar videos interactivos. En cada modelo puedes copiar el código intervenido y usar en cualquier sitio web. Es importante que mires las instrucciones, para usar la plantilla de cada modelo.

Es cierto que estos modelos, obtenidos por IA, son más sencillos de usar que los desarrollados con el editor DescartesJS; sin embargo, están limitados a presentar cuestionarios, mientras que con DescartesJS podemos tener una mayor variedad de escenas interactivas.



Videos interactivos en YouTube

En este primer modelo, es posible incluir preguntas de tipo falso o verdadero y del tipo de selección múltiple. Para el ejemplo, hemos usado tres preguntas, la primera de falso verdadero, las otras dos de selección múltiple.

Presta atención a las instrucciones que presenta la siguiente plantilla de edición HTML5, en las cuales podemos comprender cómo cambiar el video; cambiar, quitar o agregar preguntas, y cambiar la relación de aspecto del video (16:9, 9:16, 1:1, etc.)



["Configuración - Videos interactivos en YouTube"](#)

Videos interactivos en local

La única diferencia de estos modelos con los anteriores, es la fuente del video, que es en local y en formato compatible con HTML5 (mp4, ogg, WebM, etc.).



["Configuración - Videos interactivos en local"](#)

Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)

Proyecto - Chatbot con personalidad (parte 3)

Continuamos el segundo proyecto de aula, incluyendo en nuestro chatbot el texto en modo HTML, el cual nos permite ver la respuesta con el estilo HTML que queramos. Usamos scripts para incorporar los archivos de personalidad, de tal forma que sean capturadas en el navegador. Aprovechando la función `parent.set('var', var)`, generamos el historial de las conversaciones con el chatbot. En el siguiente video, puedes ver la explicación, paso a paso, de cómo implementar estas mejoras.

Guía en video. (Proyecto 2 - Parte 3)



7.4 🎬 Video - Sesión 7ª

📌 Diseño de objetos interactivos con DescartesJS e IA



📦 Tarea N°7

✦ **Video interactivo.** Usa tu creatividad y diseña un video interactivo con, al menos, tres paradas para interactuar con escenas interactivas. Usa el modelo que quieras.

Ver una posible

✓ [Solución](#)

Sesion 7. 📄 ¡Que aprendiste!



Sesión 8

Familias de gráficos

8.1 Nivel 1 - Introducción

¿Qué es un Control Tipo Gráfico?

Un control tipo gráfico en DescartesJS es un punto interactivo que puede moverse libremente por el espacio o restringirse a seguir una trayectoria específica. A diferencia de los controles numéricos, que ajustan valores mediante pulsadores o barras, los controles gráficos ofrecen una experiencia más intuitiva al permitir arrastrar elementos directamente con el ratón o en pantallas táctiles.

Para agregar un control gráfico, se oprime el botón + en el selector Controles y se selecciona la interfaz gráfico. Por defecto, los identificadores de estos controles empiezan con la letra g.

Parámetros del Control Gráfico

id: Identificador para referenciar el control (ej. g1).

espacio: Menú para escoger el espacio donde aparecerá el control.

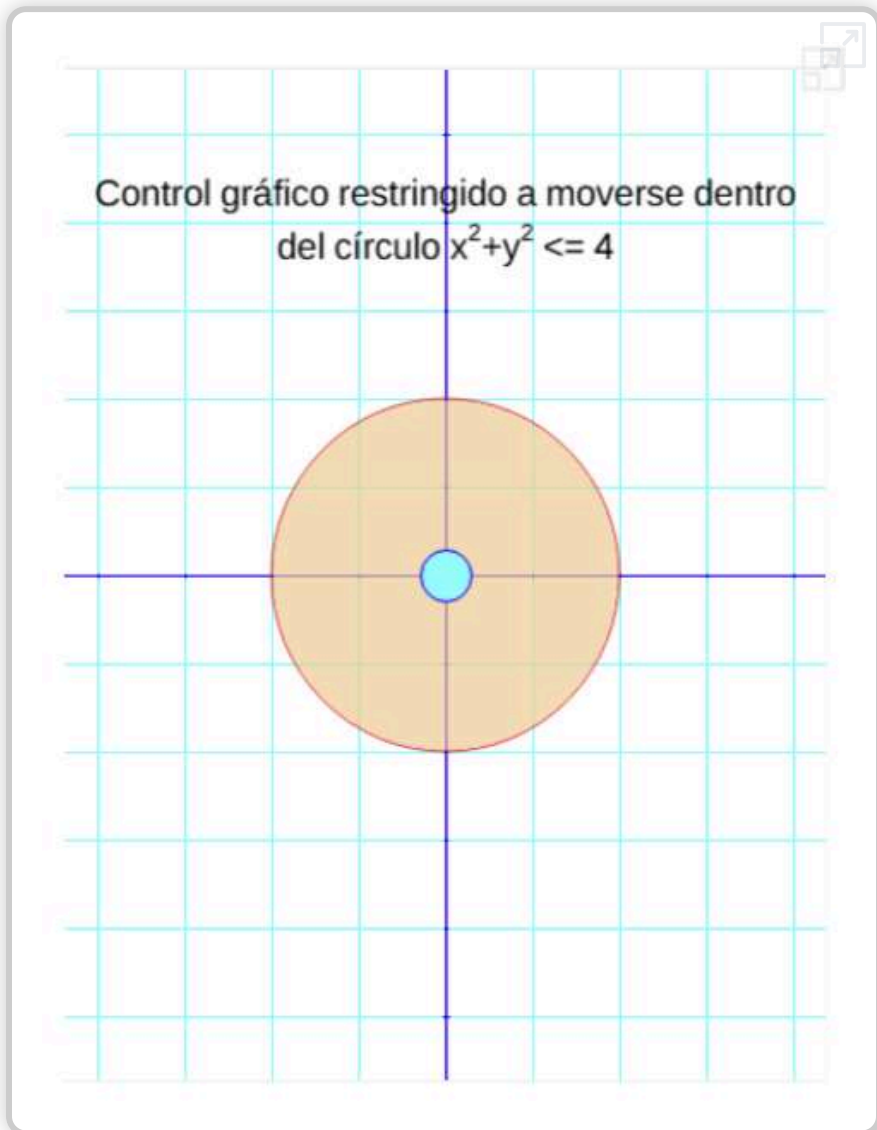
expresión: Coordenada inicial del control, ej. (0,0).

tamaño: Radio en píxeles del círculo que representa al control.

constricción: Una de las propiedades más potentes. Es un campo donde se introduce una ecuación o inecuación que restringe el movimiento del control. Esto obliga al punto a "moverse" sobre una curva o dentro de una región.

$x^2 + y^2 = 4$ restringirá el control a una circunferencia de radio 2.
 $y = x^2$ o, equivalentemente, $y - x^2 = 0$ restringirá el control a una parábola. $x^2 + y^2 \leq 4$ permitirá que el control se mueva libremente solo dentro de un círculo de radio 2 (compruébalo con el objeto interactivo de este apartado).

color, color interior, imagen: Opciones de personalización visual.



Familias de gráficos

Otra de las opciones interesantes de DescartesJS es el uso de familias para el diseño de gráficos. El funcionamiento de esta opción es similar al de un ciclo o bucle tipo Do-While. En DescartesJS, el segundo ejemplo Do-While en código JavaScript sería de la siguiente forma:

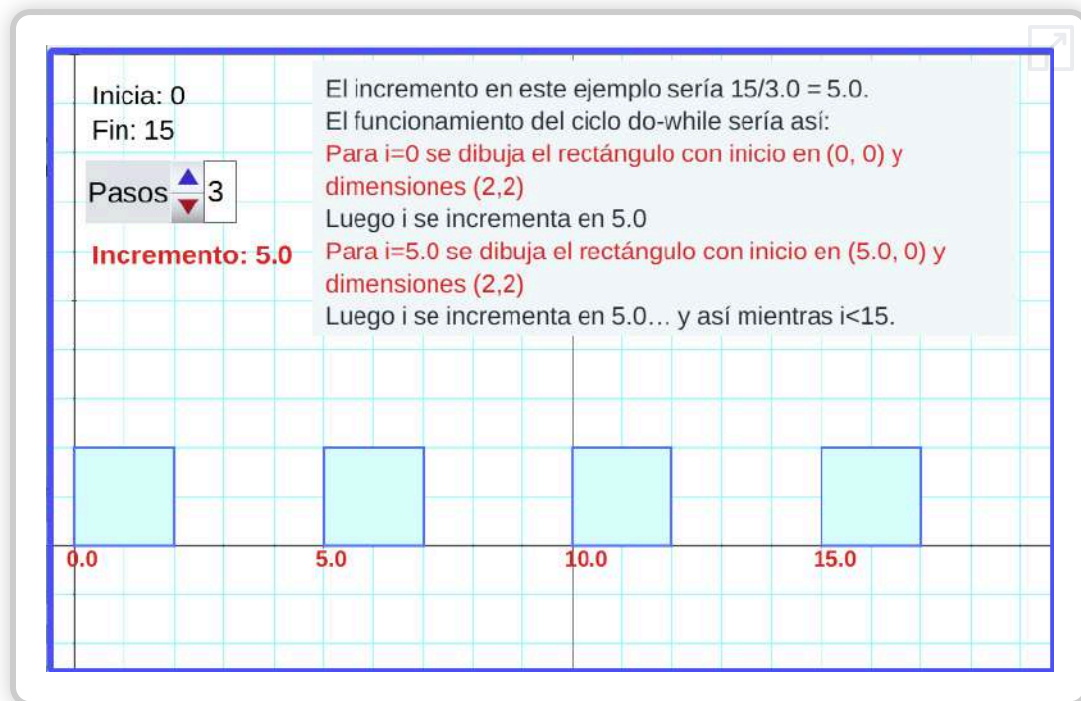
En el selector Gráficos hemos creado un rectángulo cuyas coordenadas varían con la variable **s** (observa que está activada la opción familia).



Esta variable toma valores según el intervalo, es decir, desde un valor inicial que hemos llamado **Inicia**, hasta un valor final **Fin**, el incremento depende del número de **Pasos** definidos, que se calcularía así:
Incremento = (Fin – Inicia)/Pasos.

Controles		Definiciones		Programa		Gráficos		Animación	
info Familia de rectángulos									
espacio	E1	fondo	☐	color	☐	rastros	☐	☐	☐
dibujar si								coord abs	☐
expresión (s,0,2,2)									
familia	☒	parámetro	s	intervalo	[Inicia,Fin]	pasos	Pasos		
relleno	☒				ancho	2			
radio del borde	0				estilo de línea	solida			

Veamos un ejemplo interactivo para un intervalo $[0,15]$ y diferentes valores de la variable **Pasos**:



Una de las funcionalidades de DescartesJS es la posibilidad de utilizar "familias" para la creación de gráficos, lo que permite repetir un objeto gráfico varias veces, modificando algunos de sus parámetros. Esto es útil para generar patrones, secuencias o estructuras complejas con facilidad.

En términos prácticos, esto funciona similar a un bucle en programación, donde se define una variable que actúa como parámetro para la familia de objetos, y se especifica el intervalo de valores que dicha variable tomará.

8.2 Nivel 2

Diseño de secuencias en DescartesJS

Justificación



El uso de escenas interactivas tipo secuencia (como ordenación cronológica, causa-efecto o pasos de un proceso) en educación está ampliamente justificado por su capacidad para mejorar la comprensión, la retención y la participación activa del estudiante. Estas escenas permiten representar el conocimiento de manera estructurada y dinámica, lo cual resulta especialmente útil en áreas donde el orden de los eventos, conceptos o acciones es esencial para la comprensión.

- ✓ **Favorecen el aprendizaje significativo:** Al organizar elementos en una secuencia lógica, los estudiantes relacionan la nueva información con conocimientos previos, lo que facilita la construcción de esquemas mentales más sólidos.
- ✓ **Desarrollan habilidades de pensamiento crítico y lógico:** Ordenar una secuencia implica analizar, comparar y tomar decisiones, habilidades clave en el pensamiento analítico y en la resolución de problemas.
- ✓ **Mejoran la retención y la memoria a largo plazo:** La disposición temporal o secuencial ayuda a los estudiantes a recordar con mayor facilidad procesos, etapas o hechos históricos.

Diseño de secuencias en DescartesJS - Parte 1

Ahora, vamos a nuestra actividad. Nuestra tarea es dibujar cuatro rectángulos que se constituirán en los contenedores en los cuales irán en orden secuencial las imágenes. Alguien podría decir que bastaría con dibujar cuatro rectángulos y olvidar lo de las familias, conclusión que es cierta, pero el uso de familias facilita el diseño de esta actividad, tal como lo veremos en la segunda parte.

Es muy importante que hayas practicado con la escena anterior, para comprender cómo generamos los contenedores y, además, que sepas diferenciar las coordenadas relativas de las absolutas. Observa, con cuidado, el video y empieza a diseñar tu secuencia.

 **Guía en video.**



Diseño de secuencias en DescartesJS - Parte 2

Es importante que comprendas que todo control gráfico tiene unas coordenadas asociadas. Si el control está identificado (id) como **gn**, sus coordenadas asociadas son **(gn.x,gn.y)**. El tamaño de 60 para el control gráfico, ahora es comprensible. Obviamente, no nos interesa que el control se muestre en nuestro espacio de trabajo, por ello, dejaremos un valor **cero (0) en la casilla dibujar si**, que tendrá como efecto la desaparición visual de los controles.

Por otra parte, debemos garantizar que la imagen quede incrustada en alguno de los cuatro contenedores, para lo cual recurrimos al uso de varios condicionales, que fijan las abscisas y ordenadas al centro de los cajones. Observa entonces, con mucho cuidado, el segundo video de esta actividad.

 **Guía en video.**

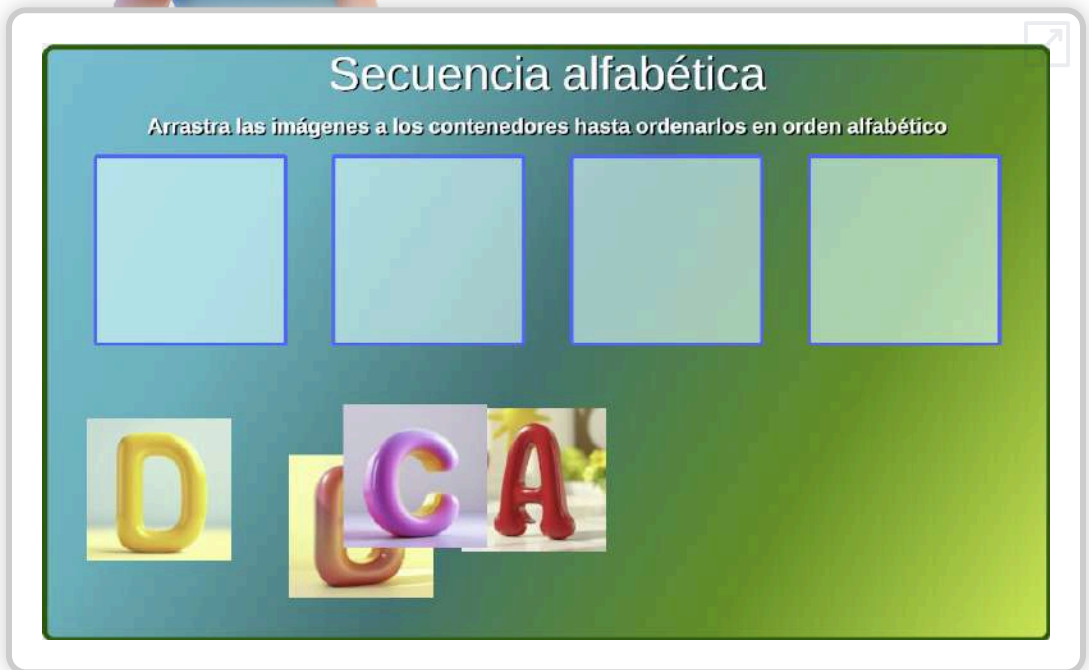


Un libro guía para diseñar el interactivo de las secuencias paso a paso en DescartesJS (pag. 201):

 "[Libro Interactivo, DescartesJS - Nivel I](#)"



A continuación, presentamos la escena interactiva obtenida según lo explicado en los dos videos.



8.3 Nivel 3

Diseño de secuencias con herramientas de IA

Herramienta 56

Caja de herramientas de IA de nuestra Red

¿Quiéres generar secuencias temporales con tres o cuatro momentos? Esta herramienta te ayudará en ello; para ello, haz clic sobre la imagen y sigue estas instrucciones:

1. Elige el número de momentos de la secuencia (3 o 4)
2. Selecciona el estilo artístico, para las imágenes
3. Incluye el tema, para que la IA genere las descripciones de cada momento, o **ingresa las descripciones personalizadas** (recomendado)
4. Cambia o acepta las imágenes generadas
5. Genera y descarga la secuencia



A continuación, presentamos dos ejemplos generados a partir de descripciones personalizadas. Es importante que sepas que la herramienta enlazada a la imagen es el modelo **premium**, usando el modelo gptimage.

Secuencia alfabética

Seguro has notado que hemos llamado esta actividad "Secuencias"; sin embargo, la herramienta 56 habla de "secuencias temporales", denominación que recoge el nombre dado en las primeras plantillas publicadas en la RED. Pero, también has notado, que su campo de aplicación va más allá de lo cronológico. Tal es el caso de la siguiente secuencia, generada con estas cuatro descripciones: Letra G, Letra H, Letra I y Letra J, con estilo Pixar



Secuencia culinaria

Para este ejemplo, usamos estas descripciones: Reunir ingredientes de cocina, Preparar los ingredientes (cortar, picar), Cocinar el plato y Servir y comer, también con estilo Pixar.



Mi primer Objeto Interactivo de Aprendizaje (OIA) con Inteligencia Artificial (IA)

Proyecto - Chatbot con personalidad (parte 4)

En esta cuarta y última parte, agregamos a nuestro chatbot dos botones de descarga del historial en formato HTML y TXT. Incluimos, además, la posibilidad de generar imágenes.

En el siguiente video, puedes ver la explicación, paso a paso, de cómo implementar estas mejoras.

Guía en video. (Proyecto 2 - Parte 4)



8.4 🎬 Video - Sesión 8ª

📌 **Diseño de objetos interactivos con DescartesJS e IA**



📦 Tarea N°8

✦ **Secuencia.** Usa tu creatividad y diseña una secuencia, usa el modelo que desees.

Ver una posible

✅ **Solución**

Sesion 8. 📄 ¡Que aprendiste!



Prácticas

Tareas y Proyectos de aula

9.1 Actividades (tareas)

9.1.1 🎯 Tarea N°1

Cuestionario

Marte: El planeta rojo 🌏

Nombre del usuario:

Cantidad de preguntas
(Min.5 - Max.30):

Activar tiempo

☐

Iniciar Cuestionario

Aciertos: 0 Fallos: 0

Cuestionario elaborado con Websim.ai

Consta de una pantalla de inicio, donde se solicita ingresar nombre del usuario, cantidad de preguntas del cuestionario (tiene una base de 30 preguntas, que se generan aleatoriamente), y una casilla de verificación si desea elaborar el cuestionario con tiempo.

Preguntas de selección múltiple con única respuesta, mostrando fallos y aciertos al responder (botón verificar) y pantalla emergente si la pregunta no es correcta.

Al finalizar el cuestionario muestra acierto, fallos y puntaje sobre 5, además, el tiempo de elaboración del cuestionario.



Botón para utilizar calculadora de conversión de unidades de longitud que se utilizan en la astronomía.

Código:

```
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Activa tu Pensamiento</title>
  <link rel="stylesheet" href="style_quiz.css"
type="text/css">
</head>

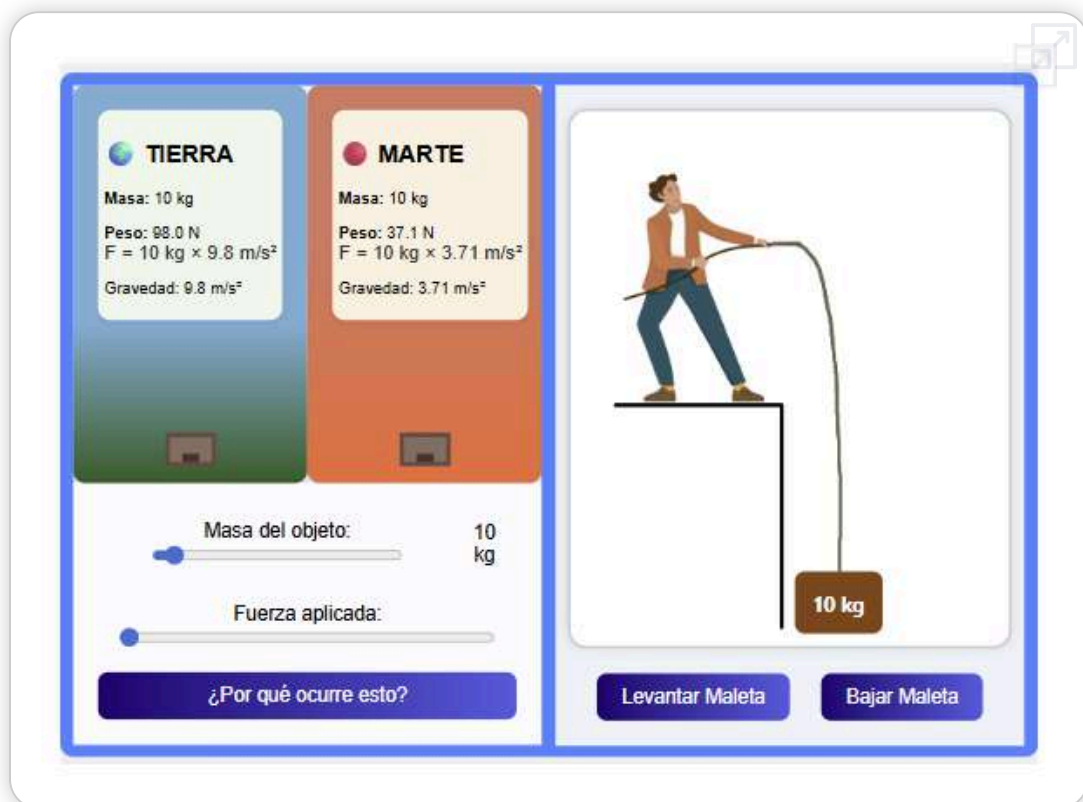
<body>
  <div class="container">

    <div class="score-board">
      <div class="timer-container" style="margin-top:140px;
margin-right:220px">
```

9.1.2 🎯 Tarea N°2

1º Ejercicio - Escena con el editor DescartesJS

El ejercicio presenta un diseño de una escena con tres espacios 2D. En dos de los espacios carga paginas HTML con animación elaboradas en **Websim.ai** y **ChatGPT**.



2º Ejercicio - Mi primer OIA con IA

Generador de imagenes con *Pollinations.ai* y *Descartes JS*

Escena interactiva con dos espacios, uno de ellos genera imágenes con *Pollinations.ai*

Chatbot Creativo

Ejercicio IA Generativa

Generación de imagenes con Pollinations.ai



Describe la imagen que deseas

Enviar



```
file='https://image.pollinations.ai/prompt/'+mensaje  
file=file+'?width=550&height=550'
```

file: variable de cadena, puede ser cualquier nombre.

mensaje: control, campo de texto. (ingresa la imagen que deseas)

En la imagen se coloca la variable **[file]**

9.1.3 🎯 Tarea N°3

2° Ejercicio (parte 2) - Mi primer OIA con IA - Mejorado

Generador de imagenes con *Pollinations.ai* y *Descartes JS*

Se combina con un presentador de diapositivas (muestra una pequeña descripción), con botones para adelantar y regresar, para generar la imagen paso a paso (relación de aspecto, estilo y Prompt).



**Ejercicio IA Generativa
(Parte 2)**

**Generación de imagenes
con Pollinations.ai**

En una imagen, el ancho y el alto están definidos por su relación de aspecto, que es la proporción entre el ancho y la altura de la imagen. Esta relación determina la forma de la imagen, por ejemplo, si es cuadrada (1:1), rectangular (3:2), o de pantalla ancha (16:9). El tamaño de la imagen se refiere a su resolución en píxeles, que es la cantidad de píxeles en el ancho y el alto.

Relación de aspecto (tamaño)

Ancho x Alto

Esquema interactivo diseñado por
Carlos Alberto Rojas Hincapié

```
images='https://image.pollinations.ai/prompt/'+mensaje+';'+E[estilos]  
images=images+'?width='+ancho+'&height='+alto
```


Vector: Para seleccionar los estilos por medio de un control *menu*.

```
E[0]=''  
E[1]='3D'  
E[2]='Barroco'  
E[3]='Acuarela'  
E[4]='Pastel'  
E[5]='Oleo'  
E[6]='Van Gogh'  
E[7]='Boceto a lapiz'  
E[8]='Arte callejero'  
E[9]='Psicodélico'
```

Configuración de la escena: Presentador de Diapositivas.

Configurar

Escena Espacios Controles Definiciones Programa Gráficos Animación

Título de la página: Chatbot IA Generativa

ancho: 978 alto: 608

botón créditos botón config botón inicio botón imprimir

filas al norte: 6 filas al sur: 0

ancha oeste: 158 ancha este: 158 alta filas: 40

signo decimal: . idioma: español mostrar región exterior expandir escena escalar

imagen del navegador

Aceptar Cerrar Aplicar

ESCENA

```
<param name="Versión" value="5.5">  
<param name="tamaño" value="978x608">  
<param name="decimal_symbol" value=".">  
<param name="expand" value="fit">  
<param name="nombre" value="">  
<param name="editable" value="no">  
<param name="Idioma" value="español">
```


9.1.4 🎯 Tarea N°4



Problema N°1 - Usando IA.

En un almacén se tiene la siguiente promoción: todo artículo con un precio superior a \$25,000 tendrá un descuento del 20%, los demás tendrán el 10%. Determinar el precio final que debe pagar una persona por comprar un artículo y el valor del descuento



Mi papelería

[Calculadora de descuento - Explicación del problema](#)

Selecciona un artículo



Precio del producto (COP)

Calcular

Problema N°2 - Usando DescartesJS.

Crea un programa en DescartesJS que reciba una calificación numérica entre 0 y 100 e imprima el nivel correspondiente:

-  90 a 100: Excelente
-  80 a 89: Notable
-  70 a 79: Aprobado
-  Menos de 70: Reprobado

Ingresa 3 notas entre 0 y 100

✓ Nota 1:

0

✓ Nota 2:

0

✓ Nota 3:

0

Promedio



```
<param name="VersiÃ³n" value="5.5">
<param name="tamaÃ±o" value="600x550">
<param name="decimal_symbol" value=".">
<param name="expand" value="fit">
<param name="nombre" value="">
<param name="editable" value="no">
<param name="Idioma" value="espaÃ±ol">
<param name="Botones" value="crÃ©ditos=no config=no inicio=no
limpiar=no filas_norte=0 filas_sur=0 ancho_este=150
ancho_oeste=150 alto=40 ">
<param name="E_01" value="tipo='R2' id='E1' ancho='100%'
alto='100%' red='no' red10='no' ejes='no' border_radius='20' ">
<param name="E_02" value="tipo='R2' id='E2' x='30' y='50'
ancho='90%' alto='60%' escala='30' fondo='8600baff' red='ee000000'
red10='eb000000' ejes='no' border_width='3' border_color='0000ff'
border_radius='20' info='Notas' ">
<param name="C_01" value="id='N1' tipo='numÃ©rico' interfaz='campo
de texto' regiÃ³n='interior' espacio='E2' nombre='Ã¡rea... Nota 1:'
expresiÃ³n='(30,30,170,40)' incr='1' min='0' max='100'
decimales='0' visible='sÃ' activo-si='muestra=0'
label_color='ffe0e4e8' label_text_color='ff0000' negrita='sÃ' ">
<param name="C_02" value="id='N2' tipo='numÃ©rico' interfaz='campo
de texto' regiÃ³n='interior' espacio='E2' nombre='Ã¡rea... Nota 2:'
expresiÃ³n='(30,100,170,40)' incr='1' min='0' max='100'
decimales='0' visible='sÃ' activo-si='muestra=0'
label_color='ffe0e4e8' label_text_color='ff0000' negrita='sÃ' ">
<param name="C_03" value="id='N3' tipo='numÃ©rico' interfaz='campo
de texto' regiÃ³n='interior' espacio='E2' nombre='Ã¡rea... Nota 3:'
expresiÃ³n='(30,170,170,40)' incr='1' min='0' max='100'
decimales='0' visible='sÃ' activo-si='muestra=0'
label_color='ffe0e4e8' label_text_color='ff0000' negrita='sÃ' ">
<param name="C_04" value="id='b4' interfaz='botÃ³n'
tipo='numÃ©rico' regiÃ³n='interior' espacio='E2' nombre='Promedio'
expresiÃ³n='(30,250,150,40)' color='ffffff' color-int='007eff'
negrita='sÃ' fuente_puntos='20' acciÃ³n='calcular'
parÃ¡metro='muestra=1' dibujar-si='muestra=0'
extra_style='overColor=0000ff|borderRadius=15' ">
<param name="C_05" value="id='b5' interfaz='botÃ³n'
tipo='numÃ©rico' regiÃ³n='interior' espacio='E2' nombre='Otra vez'
```

9.1.5 🎯 Tarea N°5

Problema 1.

Leer información de 50 registros en una hoja Excel y exportarlos a DescartesJS, que calcule:

Análisis DescartesJS Base de datos

Selecciona una ¿Pregunta?



Lectura de datos externos: Una forma sencilla es copiar los datos, pegarlos en un bloc de notas y guardarlos en una archivo txt, para luego leerlos en un vector de DescartesJS. !Importante; datos tipo cadena de caracteres, deben ir entre comillas simples.

Problema 2.

Subir el archivo al Chatbot Investiga y formula 5 preguntas; por ejemplo, ¿cuántas mujeres tienen más de 40 años?. ¿cuántos hombres tienen nombre que empiezan por vocal? o ¿cuántas mujeres mayores de 30 años, miden menos de 1,70?, etc.

 Base de datos

 Resultados chatbot

 Excel

1 / 2

−

+















Nombre	Género	Edad	Estatura
Juan Pérez	Masculino	34	1,76
María Gómez	Femenino	28	1,77
Luis Rodríguez	Masculino	41	1,55
Ana Martínez	Femenino	23	1,78
Carlos Sánchez	Masculino	55	1,63
Laura Fernández	Femenino	39	1,66
Jorge Ramírez	Masculino	47	1,82
Carmen Ruiz	Femenino	31	1,58
Diego Morales	Masculino	25	1,70
Sofía Herrera	Femenino	29	1,77
Pedro Ortega	Masculino	63	1,58
Paula Castro	Femenino	22	1,60
Andrés Navarro	Masculino	37	1,90
Elena Mendoza	Femenino	42	1,68
Tomás Aguirre	Masculino	30	1,62
Isabel Ríos	Femenino	44	1,60
Ricardo Vargas	Masculino	52	1,55
Patricia León	Femenino	33	1,62
Gabriel Salazar	Masculino	26	1,67

9.1.6 🎯 Tarea N°6

La Tierra, Día: 46

Los números del Sistema Solar

El sistema solar es una inmensa obra de ingeniería natural donde los números nos revelan sus secretos más fascinantes. Desde el diminuto Plutón hasta el gigantesco Júpiter, cada planeta tiene características únicas que pueden compararse con nuestro hogar, la Tierra. Esta herramienta interactiva está diseñada para que puedas explorar visualmente y con precisión numérica las enormes diferencias que existen entre los cuerpos celestes de nuestro vecindario cósmico.

A través de datos como el diámetro, la masa, la temperatura, la distancia al Sol y el número de lunas, esta aplicación te permite comparar directamente cualquier planeta

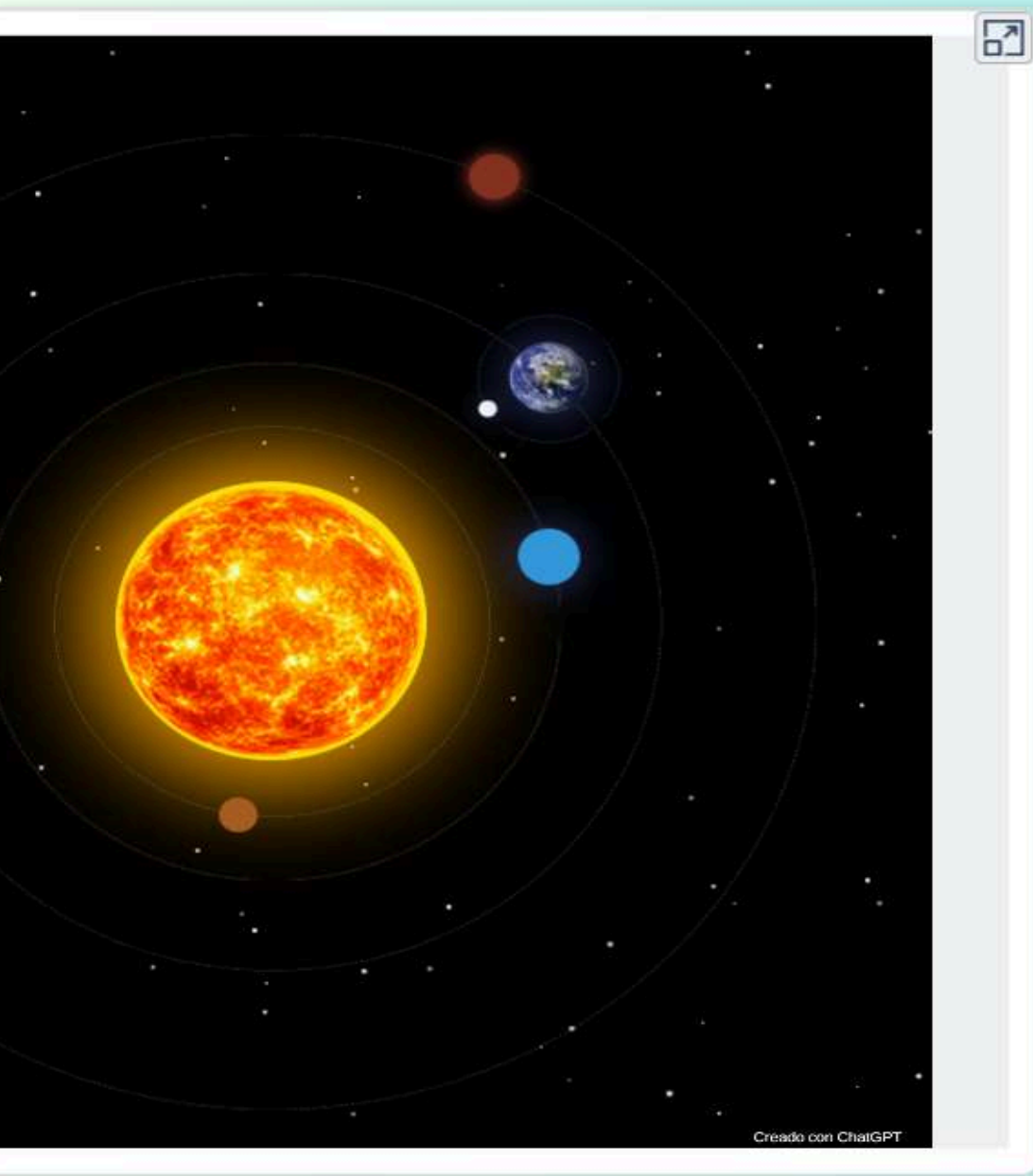
Más información...

DescartesJS

AI



DescartesJS - IA. Escenas interactiva con el editor DescartesJS y con IA, con al menos, dos tipos de controles.



9.1.7 🎯 Tarea N°7



Video Interactivo

Usando IA e incorporandolo en DescartesJS.

Video interactivo. Usa la creatividad y diseñando un video interactivo con, al menos, tres paradas para interreactuar con escenas interactivas. Usa el modelo que quieras.

Video en local utilizando la IA, capturado en [Simulación Stellarium](#)²

Codigo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-
scale=1.0"/>
  <title>La Luna - Video Interactivo</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      display: flex;
      flex-direction: column;
      align-items: center;
      justify-content: center;
      margin: 0;
      padding: 20px;
      background-color: #000;
    }
  </style>
</head>
<body>
```

² Aplicación para dispositivos móviles y computadoras que permite observar el cielo, es un planetario de código abierto para su computadora. Muestra un cielo auténtico en 3D, tal como lo que ve a simple vista, con binoculares o un telescopio. Identifica estrellas, constelaciones, planetas, satélites y otros objetos del espacio. <https://stellarium.org/es/>

Video Interactivo - Usando IA.

Video interactivo que genera las preguntas en orden aleatorio.

¿Qué sabes de la Luna?



00:00/00:00

Fecha y hora

2025 - 2 - 28 18 : 41 : 59

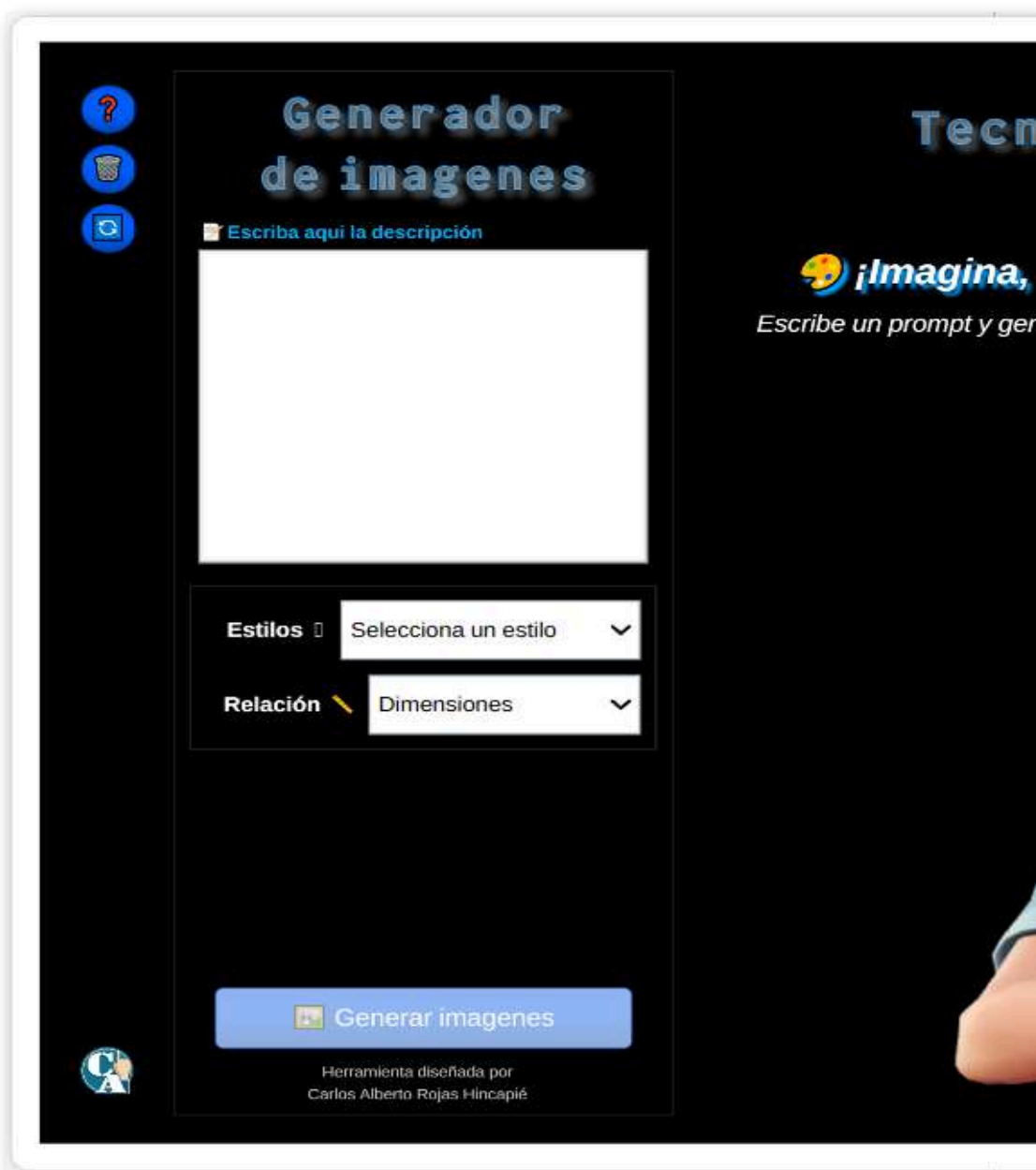
La Luna es el único satélite natural de la Tierra y el quinto satélite más grande del sistema solar. Se encuentra a una distancia promedio de 384,400 kilómetros de la Tierra y tarda aproximadamente 27.3 días en orbitarla. La Luna siempre nos muestra la misma cara debido a la rotación sincrónica con la Tierra.

FASES DE LA LUNA

En todo nuestro sistema solar, el único objeto que brilla con luz propia es el

9.2 Proyectos

Generador de imágenes



9.2.1 🧑🏫 Proyecto de aula 1

**tecnología Pollinations
y DescartesJS**

Escribe y Crea con Tecnología Pollinations!

Genera imágenes con estilos y dimensiones únicas. ¡Tu creatividad es el límite!



9.2.2 🧑📖 Proyecto de aula 2



Chatbot Descartes

 **Personaje**

Académico 

 **Modelo**

GR

 Escriba aquí la descripción y  enviar

 **Enviar**

  **Descargar .html**

 **Descargar**

Chatbot con personalidad

Tecnología Pollinations y DescartesJS

