

De los escape rooms a los SCORM autoevaluables

Cómo crear materiales interactivos
para Física y Química con ayuda de la IA



Jueves 15/01/26

Sesión 1: "De escape room a SCORM"

```
Linux: Linux
devicePrefixes {
  iPhone: 'iPhone',
  iPod: 'iPod',
  iPad: 'iPad'
},
maxIVersion: 12,
/
* The default function that detects various platforms and sets tags
* in the platform map accordingly. Examples are iOS, android, tablet, etc.
* @param tags the set of tags to populate
*/
detectPlatformTags: function() {
  var me = this, ua = navigator.userAgent,
  isMobile = /mobile(\\|\\s)/.test(ua),
  element = document.createElement('div'),
  isEventSupported = function(name, tag) {
    if (tag === undefined) {
      tag = window;
    }
    var eventName = name.toLowerCase();
    var eventSupported = (eventName in element);
    if (!eventSupported) {
      element.setAttribute(eventName, 'x');
      eventSupported = !element.getAttribute(eventName);
    }
    if (eventSupported) {
      if (matched && (matched === 'HTC' || matched === 'version' && matched === 'HTC')) {
        version = match && match.length > 1 ? parseFloat(match[match.length - 1]) : 0;
      } else {
        version = match && match.length > 1 ? parseFloat(match[match.length - 1]) : 0;
      }
      version = match && match.length > 1 ? parseFloat(match[match.length - 1]) : 0;
    }
    if (version) {
      activeCount++;
    }
    systems[key] = version;
  }
  systems[key] = version;
  keys = _getKeys(me.fallbackOSPrefixes);
  // If we could not find the fallbacks, otherwise we just
  // falsify the fallbacks
  len = keys.length;
  for (index = 0; index < len; index++) {
    key = keys[index];
    // If we detected from osPrefixes
    if (activeCount === 0) {
      value = me.fallbackOSPrefixes[key];
      match = ua.toLowerCase().match(new RegExp(value));
      systems[key] = match ? true : 0;
    } else {
      systems[key] = 0;
    }
  }
  return systems;
}
```

Objetivos de la sesión



Evolución

Mostrar la **evolución completa** desde mis primeros escape rooms hasta los recursos SCORM autoevaluables, destacando los hitos clave y lecciones aprendidas en el camino.



Ejemplos reales

Compartir **pantallas y mecánicas reales** de los recursos desarrollados, con capturas de juegos, actividades interactivas y feedback visual.



Plantillas IA

Ofrecer **plantillas de prompts** probados y listos para usar, con ejemplos específicos para diferentes tipos de recursos interactivos.



Integración en Aules

Explicar paso a paso cómo **integrar estos recursos en Aules** (Moodle/LMS) de forma efectiva, aprovechando todas las funcionalidades de seguimiento y evaluación.



SCORM propio

Conseguir que cada docente salga con su **SCORM personalizado** listo para usar en sus clases, adaptado a sus necesidades específicas y contenidos.

Primera idea: Escape Rooms Didácticos



¿Por qué escape rooms?

- **Soporte visual y lúdico** en unidades de Física y Química
- **Trabajo en equipo** y evaluación de competencias
- **Aprendizaje significativo** a través de la gamificación



Ambientación: Rick and Morty



Atracción visual



Humor



Ciencia



Creación con IA

Voces

Generadas con IA

Imágenes

Generadas con IA

Ejemplo de pantalla de inicio



Archivo principal

<https://www.fernandoharo.es/escape1/index.html>

Página de inicio del escape room con elementos multimedia y navegación intuitiva.

Contenidos del archivo:



Video de presentación

Archivo mp4 integrado



Texto narrativo

Introducción a la actividad



Botón para comenzar

Enlace a la siguiente pantalla



Responsive

Tablet y móvil



Autocontenido

Con conexión externa



Oh no, ¡los elementos están desapareciendo! ¡Esto podría desestabilizar todo el universo!
¡Debo arreglarlo antes de que sea demasiado tarde! Mientras más elementos se
desvanecen, Rick comienza a trabajar frenéticamente en sus dispositivos, tratando de
evitar que la tabla periódica desaparezca por completo.

¡Morty, eh, quiero decir, estudiantes! ¡Estoy perdiendo la cabeza aquí! ¡La tabla periódica se ha vuelto
local! ¡Ayúdenme a resolver esto antes de que implote la realidad!

Iniciar la aventura

Pantallas con áreas interactivas

Ejemplos concretos



https://www.fernandoharo.es/escape1/inside_lab.html
Laboratorio de Mendeleev con elementos interactivos



<https://fernandoharo.es/escape3/drogueria.html>
Tienda antigua con compuestos químicos

Problema técnico

Problema

La IA no siempre acierta las coordenadas de los **map area**

Solución

Ajuste manual en el código:

```
data-coords="455,545,589,715"
```



Sistema de coordenadas

(0,0)

Esquina superior izquierda

Máx

Ancho × Alto imagen

> Prompt para generar este recurso

Genera una página HTML con un mapa de imagen interactivo. La imagen es una droguería antigua. Al hacer clic en cada frasco, debe mostrarse un alert con el nombre del compuesto y su fórmula.

Usa solo HTML, CSS y JavaScript. La imagen debe ser responsive. Incluye un botón para resolver un enigma con 10 preguntas tipo test.

Requisitos técnicos:

- Mapa de imagen con áreas clickeables
- Eventos onclick en cada zona
- Diseño responsive
- Sistema de evaluación integrado

Pantallas con drag & drop



Ejemplo práctico

https://fernandoharo.es/escape5/laboratorio_abandonado.html



Etiquetas arrastrables

Con nombres de compuestos químicos



Clasificación

Orgánicos vs inorgánicos



Verificación automática

Feedback inmediato

💡 Mecánica pedagógica

El usuario debe **arrastrar cada etiqueta** a la zona correcta (orgánica o inorgánica). Al soltar, el sistema verifica automáticamente y proporciona **feedback visual inmediato** (verde/rojo). Una vez completado, muestra puntuación y permite reintentar.

</> Tecnología

HTML5 API

Drag and Drop nativo

Eventos JavaScript

ondragstart, ondrop, ondragover

CSS Visual

Transiciones y estados hover

Ventajas

- ✓ Interacción táctil intuitiva
- ✓ Refuerza el aprendizaje activo
- ✓ Evaluación inmediata
- ✓ Compatible con tablets

Juegos dinámicos



https://fernandoharo.es/escape3/lluvia_valencias.html

Juego arcade

Elementos químicos cayendo. El usuario debe capturar los que tienen la valencia objetivo. Incluye puntuación en tiempo real y efectos de sonido.



Control: mouse/táctil



https://fernandoharo.es/escape3/juego_cartas.html

Formar fórmulas

Sistema de cartas donde los estudiantes combinan elementos para formar compuestos correctos. Modo multijugador opcional.



1-4 jugadores



Próximamente

Nuevos juegos

• Bingo de formulación:

<https://fernandoharo.es/escape5/bingo.html>

• Oca de orgánica

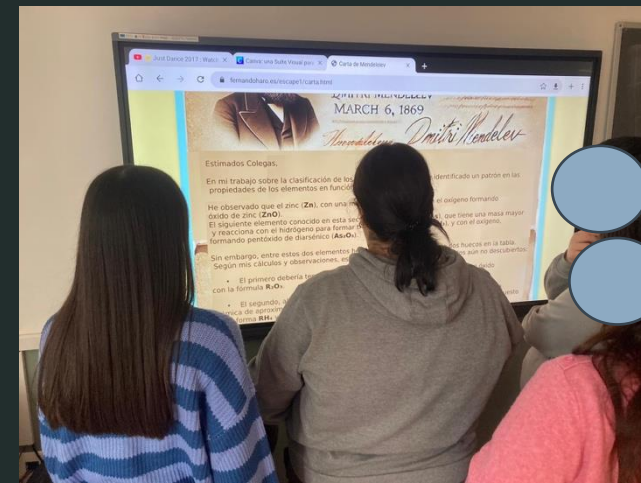
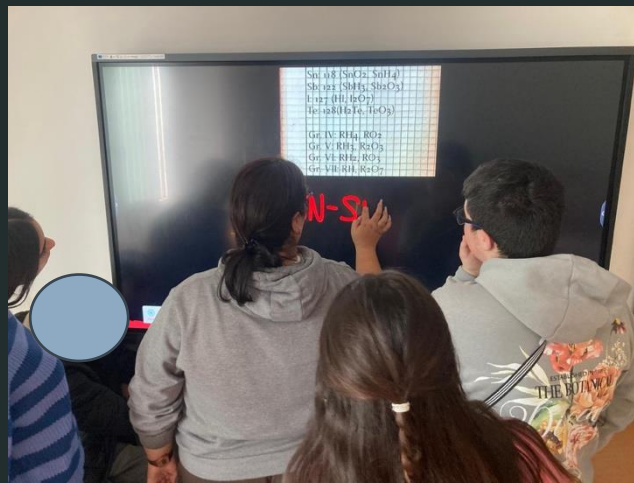
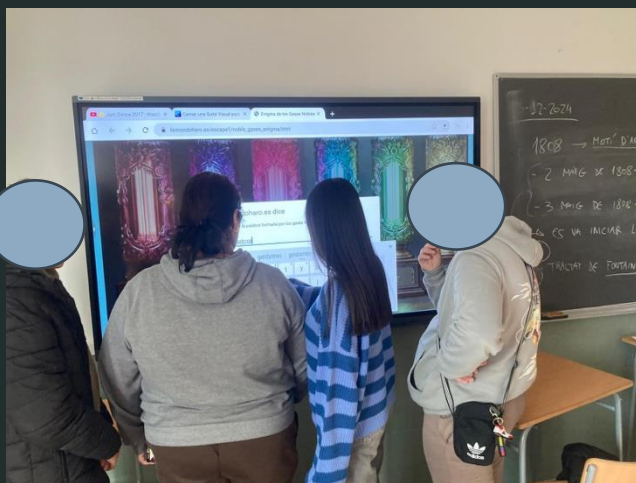
<https://fernandoharo.es/escape5/oca.html>

• Simulador CanSat

<https://fernandoharo.es/escape5/paracaidas.html>



Feedback del alumnado



El problema del servidor personal



¿Qué pasó?

Los primeros recursos funcionaban en un **servidor VPS (Virtual Private Server)**.



Coste económico

No lo debe soportar el docente.



Bloqueos en la red

Filtrado de contenido externo, podría suceder



No evaluable

No se integra en aulas

Resultado

Viable pero eepende de la política del departamento de IT del IES y consellería



¿Solución?

Migrar a un sistema **integrado en la plataforma educativa** oficial del centro.



SCORM integrado

Paquete ZIP autocontenido que se sube directamente a Aules



Autoevaluable

Corrección automática con tolerancia en respuestas numéricas



Envío de notas

Puntuación automática en el libro de calificaciones



Resultado: Recursos estables, seguros y totalmente integrados en el flujo de trabajo del profesorado

¿Qué es un SCORM?

Paquete ZIP con estructura específica



index.html

Contenido principal del recurso



imsmanifest.xml

Estructura y metadatos del paquete



script.js

Lógica de interacción y evaluación



style.css

Diseño y estilos visuales

Cómo se usa

1

Subir a Aules

Como recurso SCORM

2

Configurar

Intentos, puntuación, retroalimentación

3

Listo para usar

Los estudiantes acceden directamente

Funcionalidades



Intentos



Retroalimentación



Puntuación



Guardado

Ejemplo real: Mono paracaidista



<https://fernandoharo.es/escape5/mono.html> (SCORM)

Simulación interactiva

Recurso SCORM 1.2 que simula la **conservación de la energía mecánica** mediante un mono que se lanza en paracaídas desde diferentes alturas.



Slider interactivo

Control para ajustar altura inicial del mono (0-100m)



Gráficos tiempo real

Visualización de energía potencial, cinética y total



Puntuación automática

Al finalizar, envía nota al LMS automáticamente



Múltiples intentos

Los estudiantes pueden experimentar con diferentes valores



Conceptos trabajados

- Energía potencial gravitatoria
- Energía cinética
- Conservación de energía
- Transformación energética
- Rozamiento y disipación

Impacto en el aula

+90%

Comprensión del concepto

"Los estudiantes visualizan el principio de conservación de energía de forma interactiva, pudiendo manipular variables y observar los resultados inmediatamente."



Responsive



Online



SCORM 1.2

Prompt para generar un SCORM autoevaluable

>_Prompt completo y detallado

Actúa como desarrollador de un recurso SCORM 1.2 autocontenido. Necesito un archivo ZIP con: index.html, (script.js, style.css, embebidos en index.html) y imsmanifest.xml.

Recurso: [tema/UD]. Público: [curso]. Idioma: español con acentos correctos.

Actividad: [tipo: test/banco de problemas/mini-escape].

Requisitos:

- Aleatoriedad: [n] ítems de un banco de [tamaño].
- Corrección automática con tolerancia si hay números.
- Nota 0-100 y estado completado/aprobado con umbral [X].
- Botón "Nuevo intento" y feedback inmediato.
- Diseño claro y usable en tablet (botones grandes).

Devuélveme el contenido completo de cada archivo, listo para pegar y zippear.

✍ Campos a personalizar

[tema/UD]

Unidad didáctica o tema específico

[curso]

Nivel educativo (ESO, Bachillerato...)

[tipo]

test | banco de problemas | mini-escape

[n] y [tamaño]

Nº de preguntas y tamaño del banco

[X]

Umbral de aprobación (ej: 50)

Resultado esperado

- ✓ 2 archivos listos para usar (css y js embebidos en el index.html)
- ✓ Código limpio y comentado
- ✓ Diseño responsive
- ✓ Funcionalidad completa

```
1 mi_scorm.zip
2 |— imsmanifest.xml      (obligatorio)
3 |— index.html          (entrada del SCO)
4 |— media/              (imágenes, audio, vídeo)
5 |— (opc.) css/js       (si no va embebido)
```

Trabajo tutorizado

Cada docente creará su propio SCORM

- 1 Elegir tema**
Seleccionar unidad didáctica específica
- 2 Generar banco de ítems**
Usar IA para crear preguntas variadas
- 3 Pegar en VS Code**
Editar y personalizar el código
- 4 Guardar como .html**
Archivo principal del recurso
- 5 Crear imsmanifest.xml**
Estructura SCORM obligatoria
- 6 Comprimir en ZIP**
Paquete SCORM listo
- 7 Subir a Aules**
Publicar y compartir con el alumnado

Herramientas necesarias

ChatGPT / Claude /Kimi
Para generar el código

VS Code / Editor
Para editar y guardar

Sistema operativo
Compresor de archivos ZIP

Enfoque hands-on

En esta sesión, cada docente trabajará en **su propio recurso**, recibiendo ayuda personalizada y resolviendo dudas en tiempo real.



Resultado garantizado
SCORM propio al finalizar

Checklist de calidad



7 criterios esenciales

Todo SCORM de calidad debe cumplir estos estándares mínimos para garantizar una experiencia de usuario óptima.



Certificación de calidad

Todos los ítems deben estar marcados

1

Acentos correctos

Codificación UTF-8

2

Botones grandes

Tamaño táctil óptimo

3

Retroalimentación inmediata

Respuesta tras cada acción

4

Nuevo intento

Botón de reinicio visible

5

Aleatoriedad

Ítems diferentes por intento

6

Nota visible

Puntuación clara al finalizar

7

Compatible con tablet / móvil

Tamaños de fuente adecuados y navegación táctil

Recursos y próximos pasos



GitHub

Repositorio público con todas las **plantillas, ejemplos y recursos** utilizados en esta sesión, disponibles de forma gratuita.



Plantillas SCORM

Listas para personalizar



Juegos y escape rooms

Ejemplos funcionales



Prompts colección

Para diferentes tipos de recursos



Documentación

Guías paso a paso



Próxima sesión

STEM con IA: nuevas herramientas para el aula de Física y Química



Medida de temperatura

Sensores + visualización de datos en tiempo real



Código para Arduino/ESP32

Programación de microcontroladores con IA



Exportar datos

Generación automática de informes y gráficas



Comunidad de práctica

Compartiremos experiencias, recursos y mejoras en el repositorio común. **El aprendizaje continuo** es clave para el éxito de estas herramientas.



Gracias

¿Preguntas?



Contacto

f.haromartinez@edu.gva.es



GitHub

https://github.com/Ferhama/recursos_scom

Recursos disponibles para seguir aprendiendo

Anexo 1 : Prompt pantalla interactiva (mapas)



Actúa como desarrollador web educativo.

Quiero una actividad tipo “escape room” en una única página HTML (autocontenida: CSS y JS embebidos).

ESCENA

- Título: Laboratorio abandonado
- Estética: misterio/ciencia, pero apta para alumnado.
- Fondo: usar una imagen de laboratorio (pon un placeholder: background.jpg).
- La página debe funcionar aunque no exista la imagen (mostrar un fondo degradado).

MECÁNICA (DRAG & DROP)

- En la escena hay 8 etiquetas (chips) que el alumno debe arrastrar y colocar sobre zonas correctas del fondo.
- Ejemplo de etiquetas (editable): “matraz”, “probeta”, “balanza”, “reactivo A”, “reactivo B”, “campana”, “termómetro”, “pictograma”
- Cada etiqueta tiene un “objetivo” (target) definido por un rectángulo (x,y,w,h) en coordenadas NORMALIZADAS 0–1000.
- Al soltar una etiqueta:
 - Si cae dentro de su target → se “engancha” (snap), queda fija y muestra feedback .
 - Si no → vuelve a su posición inicial y muestra feedback .
- Debe haber un botón “Mostrar/Ocultar modo DEBUG” que dibuje los rectángulos target sobre la imagen para ajustar.

RESPONSIVE + USABILIDAD

- Mobile-first: botones grandes, arrastre cómodo con dedo.
- La escena debe mantener proporción (sin recorte) y centrarse en pantalla.
- El redimensionado debe recalcular posiciones/targets para que el snap sea correcto en móvil y en pantalla de clase.

EVALUACIÓN

- Contador de aciertos y porcentaje (0–100).
- Umbral de victoria: 70%.
- Botones: “Reiniciar” y “Nuevo intento” (recoloca etiquetas y opcionalmente cambia el orden).
- Al completar todo: mensaje de éxito y “código final” de 4 dígitos (generado al azar al iniciar, se muestra al ganar).

ENTREGA

- Devuélveme SOLO un archivo: index.html con todo dentro (CSS y JS embebidos).
- Código comentado y variables fáciles de editar:
 - lista de etiquetas
 - targets (x,y,w,h en 0–1000)
 - umbral
 - texto de la historia

Anexo 2 : Links

Visual Studio Code

<https://code.visualstudio.com/download>

GitHub

https://github.com/Ferhama/recursos_scorm

IDE Arduino

<https://www.arduino.cc/en/software/>

Web Fernando Haro

<https://fernandoharo.es>

Kimi

<https://www.kimi.co>