

UD02UT04

Manipulación del DOM

Índice

Resultado de aprendizaje, contenido y criterios de evaluación.....	1
Instrucciones.....	3
Actividades.....	3
Apartado 1 · Panel del dispositivo y navegador.....	3
Apartado 2 · Laboratorio de ventanas.....	4
Apartado 3 · Temporalización de alerta.....	4
Rúbrica.....	5

Resultado de aprendizaje, contenido y criterios de evaluación

Resultado de aprendizaje
RA3. Escribe código, identificando y aplicando las funcionalidades aportadas por los objetos predefinidos del lenguaje.
Contenido
7 - Mecanismos del navegador para el almacenamiento y recuperación de información. 8 - Depuración y documentación del código.
Criterios de evaluación
a) Se han identificado los objetos predefinidos del lenguaje c) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para cambiar el aspecto del navegador y el documento que contiene d) Se han generado textos y etiquetas como resultado de la ejecución de código en el navegador. e) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para interactuar con el usuario h) Se ha depurado y documentado el código

Resultado de aprendizaje
R4. Programa código para clientes web analizando y utilizando estructuras definidas por el usuario
Contenido
5 - Creación de objetos. 6 - Definición de métodos y propiedades
Criterios de evaluación
a) Se han clasificado y utilizado las funciones predefinidas del lenguaje b) Se han creado y utilizado funciones definidas por el usuario. g) Se ha creado código para definir la estructura de objetos. h) Se han creado métodos y propiedades k) Se ha depurado y documentado el código

Resultado de aprendizaje
RA5. Desarrolla aplicaciones web interactivas integrando mecanismos de manejo de eventos
Contenido
2 - Utilización de formularios desde código. 3 - Modificación de apariencia y comportamiento. 4 - Validación y envío. 5 - Expresiones regulares. 6 - Prueba y documentación del código
Criterios de evaluación
a) Se han reconocido las posibilidades del lenguaje de marcas relativas a la captura de los eventos producidos b) Se han identificado las características del lenguaje de programación relativas a la gestión de los eventos c) Se han diferenciado los tipos de eventos que se pueden manejar d) Se ha creado un código que capture y utilice eventos f) Se han validado formularios web utilizando eventos h) Se ha probado y documentado el código

Resultado de aprendizaje
RA6. Desarrolla aplicaciones web analizando y aplicando las características del modelo de objetos del documento
Contenido
4 - Programación de eventos 5 - Diferencias en las implementaciones del modelo 6 - Independencia de las capas de implementación de aplicaciones web
Criterios de evaluación
a) Se ha reconocido el modelo de objetos del documento de una página web b) Se han identificado los objetos del modelo, sus propiedades y métodos. c) Se ha creado y verificado un código que acceda a la estructura del documento d) Se han creado nuevos elementos de la estructura y modificado elementos ya existentes. e) Se han asociado acciones a los eventos del modelo h) Se han independizado las tres capas de implementación (contenido, aspecto y comportamiento), en aplicaciones web

Agrupamiento

En pareja

Instrucciones

- Desarrolla cada actividad.
- Comprime el proyecto.
- Nombra el fichero como TuNombre_Apellido1_Apellido2_ UD02UT03.zip

Actividades

Apartado 1 · Lucha de personajes de “El Señor de los Anillos”.

Redacta el código JS para construir una mini-app que simule una lucha entre dos personajes del universo de “El Señor de los Anillos”. El usuario elige ejército y arma para cada jugador. La app debe mostrar vida, registrar ataques y declarar ganador.

Se aporta html y css.

The screenshot shows a web application interface for a combat simulation. At the top, a dark blue header bar contains the text "Combate preparado". Below this, the interface is divided into two main columns for "Jugador 1" and "Jugador 2".

Jugador 1: Features a dropdown menu for "Ejército" with "Elfos" selected, another dropdown for "Arma" with "Espada" selected, a green progress bar representing health, the text "100 / 100", and a button labeled "Atacar".

Jugador 2: Features a dropdown menu for "Ejército" with "Orcos" selected, another dropdown for "Arma" with "Hacha" selected, a green progress bar representing health, the text "130 / 130", and a button labeled "Atacar".

Below the player columns is a section titled "Registro de batalla". It contains a large text area with the placeholder text "Registro vacío." and two buttons at the bottom: "Iniciar / Reiniciar" (highlighted in green) and "Limpiar registro".

Crea la estructura:



Reglas del juego

- Cada jugador elige ejército y arma en su panel.
- Vida inicial según ejército. Un ataque resta vida según el arma. Posible crítico.
- Gana quien deja al rival a 0. Empate si ambos llegan a 0.
- El registro muestra cada acción. Botón “Iniciar/Reiniciar” resetea. “Limpiar registro” borra las líneas.

Rúbrica

Criterio (RA)	Excelente (10)	Bien (7)	Suficiente (5)	Insuficiente (0)
RA3-a (0,7 pts) Identifica objetos predefinidos: window, document, screen, navigator, location.	Enumera y explica propósito y propiedades clave con ejemplo de uso.	Omite 1–2 objetos o da explicaciones parciales.	Lista objetos sin explicar bien propiedades ni ejemplos.	Confunde u omite la mayoría.
RA3-c (0,7 pts) Cambia aspecto del documento con DOM.	Modifica estilos y clases de forma contextual y reversible.	Cambios correctos pero limitados o sin revertir.	Cambios puntuales con errores menores.	No modifica o rompe estilos.
RA3-d (0,7 pts) Genera textos y etiquetas desde código.	Crea y elimina nodos dinámicos con contenido correcto y seguro.	Crea nodos correctos pero eliminación o contenido mejorable.	Solo añade o solo elimina; uso limitado.	No crea ni gestiona nodos.
RA3-e (0,7 pts) Interacción con usuario mediante eventos.	Captura eventos y actualiza UI sin fallos visibles.	Funciona con pequeños fallos de estado.	Responde parcialmente a eventos.	No responde a eventos.
RA3-h (0,2 pts) Depura y documenta.	Comentarios claros, console útil, pasos de prueba.	Comentarios escuetos o logs excesivos.	Comentarios mínimos, logs confusos.	Sin comentarios ni evidencias.
R4-a (0,2 pts) Usa funciones predefinidas del lenguaje.	Selección adecuada y justificada de funciones.	Selección correcta pero sin justificar.	Uso básico sin criterio claro.	Uso incorrecto o nulo.
R4-b (0,2 pts) Crea y usa funciones propias.	Funciones cohesionadas, parámetros claros, reutilizables.	Alguna función hace demasiado o nombres mejorables.	Funciones largas o acopladas al DOM.	Lógica sin funciones.
R4-g (0,2 pts) Define estructura de objetos.	Objetos bien modelados para el dominio.	Objetos correctos con leves redundancias.	Estructura mínima funcional.	Sin objetos o incoherentes.
R4-h (0,2 pts) Crea métodos y propiedades.	Métodos pertinentes y bien nombrados.	Algún método redundante o mal nombrado.	Métodos mínimos, poco cohesionados.	Sin métodos útiles.
R4-k (0,2 pts) Depura y documenta (módulo de dominio).	Comentarios de intención y pruebas básicas.	Comentarios escuetos.	Comentarios mínimos.	Sin documentación.
RA5-a (0,4 pts) Reconoce captura de eventos en HTML/JS.	Elige y aplica el mecanismo adecuado.	Aplica mecanismo válido con pequeñas carencias.	Aplica solo uno sin comparar.	Desconoce mecanismos.
RA5-b (0,4 pts) Identifica gestión de eventos en JS.	Usa addEventListener y maneja fase/burbujeo cuando procede.	Usa addEventListener sin matices.	Mezcla inline y JS o confunde alcances.	Gestión deficiente o inexistente.
RA5-c (0,4 pts) Diferencia tipos de eventos.	Selecciona tipos idóneos y justifica.	Selección correcta sin justificar.	Usa pocos tipos o inadecuados.	Tipos erróneos.
RA5-d (0,5 pts) Captura y usa eventos.	Flujo completo y estable para botones y selects.	Flujo correcto con pequeños fallos.	Parcial o con estados incoherentes.	No funcional.
RA5-f (0,4 pts) Valida formularios/entradas con eventos.	Valida selects/inputs y muestra feedback.	Valida parcialmente sin feedback claro.	Validación mínima.	Sin validación.
RA5-h (0,4 pts) Prueba y documenta eventos.	Casos de prueba y notas en código.	Pruebas básicas.	Pocas pruebas.	Sin pruebas.
RA6-a (0,6 pts) Reconoce el DOM de la página.	Mapa estructura y selecciona nodos eficientemente.	Selección correcta con redundancias.	Selecciones poco óptimas.	Selección errónea.
RA6-b (0,6 pts) Identifica objetos del	Usa correctamente classList, style, attributes, children,	Uso correcto con pequeños errores.	Uso limitado o inseguro.	Uso incorrecto.

Criterio (RA)	Excelente (10)	Bien (7)	Suficiente (5)	Insuficiente (0)
modelo, propiedades y métodos.	etc.			
RA6-c (0,6 pts) Accede a la estructura del documento.	Usa id, clase, tag, querySelector y querySelectorAll.	Usa ≥ 4 vías correctamente.	Usa 2–3 vías.	Usa ≤ 1 vía.
RA6-d (0,6 pts) Crea y modifica elementos existentes.	Crea, inserta, reemplaza y elimina con seguridad.	Crea y elimina con detalles mejorables.	Solo crea o solo elimina.	No modifica estructura.
RA6-e (0,5 pts) Asocia acciones a eventos del modelo.	Asociación clara y desacoplada del HTML.	Asociación válida con algún inline residual.	Mezcla enfoques y acopla UI.	No asocia correctamente.
RA6-h (0,6 pts) Independiza capas (contenido/estilo/comportamiento).	HTML y CSS separados; JS en ES Modules.	Separación casi completa.	Separación parcial o sin módulos.	Mezcla capas.

	Se retrasa más de 5 días	Se retrasa entre 4 y 5 días	Se retrasa entre 2 y 3 días	Se retrasa 1 día.	Entrega en plazo
Entrega tarea en plazo	- 5	- 3	-2	-1	0