



I.E.S. SERRA PERENXISA



DWEC 2º DAW

UDAD 5 EVENTOS: CONTROL GASTOS SEMANAL

INTRODUCCIÓN RESUMEN

Partiendo de los ficheros .html y .css (bootstrap) de una página web de control de presupuestos, daremos vida al mismo.

Se pedirá un presupuesto inicial al usuario y podrá ir introduciendo sus gastos, de forma que podrá ir viendo cuánto dinero le queda del presupuesto inicial.

Según vaya el usuario introduciendo gastos, éstos se incorporarán en la parte de la derecha (arriba de "presupuesto").

Los gastos se podrán eliminar.

El valor de "restante" se irá recalculando según se vayan añadiendo/borrando los gastos.

La etiqueta "restante" cambiará de color según vaya subiendo el importe de los gastos.

Gasto Semanal

Añade tus gastos aquí

Listado

Gasto:

Cantidad:

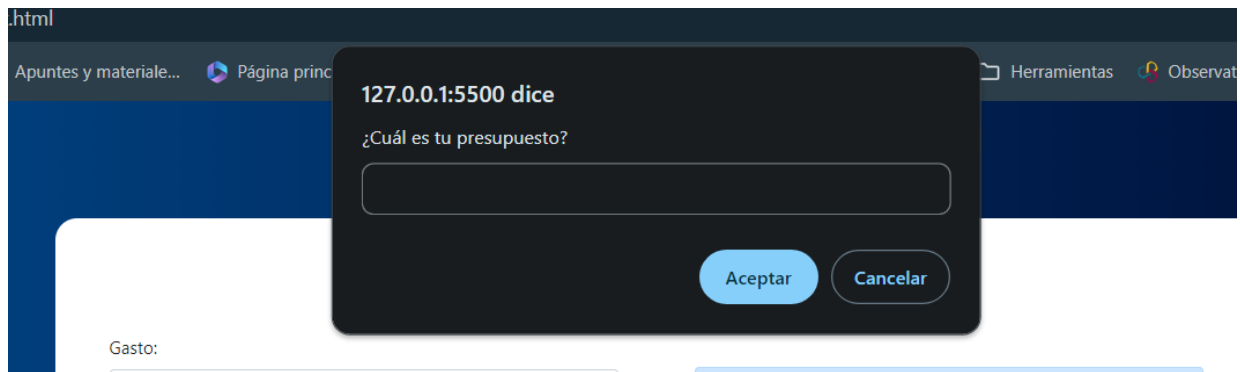
Agregar

Presupuesto: €

Restante: €

1. EJECUCIÓN: PEDIR Y EVALUAR PRESUPUESTO

Cuando la página esté cargada, se pedirá al usuario el presupuesto fijado para esa semana. Se deberá realizar las validaciones oportunas y se seguirá pidiendo hasta que el usuario ponga un importe correcto.



2. EJECUCIÓN: GESTIÓN DE LA PÁGINA Y SU INFORMACIÓN

Se deben hacer dos clases:

- **Presupuesto:** Es el presupuesto que va a poner el usuario. Controlará el importe restante y los gastos (todo lo relacionado con el presupuesto se deberá gestionar desde la clase). Su constructor recibirá un importe y tendrá 3 propiedades (presupuesto, restante y gastos). Este último será un array que contendrá los gastos que el usuario vaya introduciendo.
- **Interfaz de usuario:** Mostrará el HTML de los gastos y se encargará también de las validaciones (todo lo relacionado con lo que se añade al HTML se deberá gestionar desde la clase). Esta clase no tiene constructor ni propiedades, sólo métodos.

3. EJECUCIÓN: PRIMEROS VALORES

Al inicio, cuando el usuario ya ha introducido el presupuesto, este importe se debe reflejar tanto en "presupuesto" como en "restante"



4. EJECUCIÓN: VALIDACIÓN DEL FORMULARIO

Se deben rellenar correctamente los dos campos del formulario que permiten añadir el gasto son obligatorios. Si no pasa la validación, mostraremos un mensaje de alerta

Gestión de las alertas o mensajes: Los mensajes de alerta los crearemos con DOM Scripting, con las clases de Bootstrap de “text-center” y “alert”.

Esto lo haremos en una función **reutilizable** para gestionar las alertas (hay alertas de error y de información positiva). En este caso, es de error, por lo que al elemento le añadimos la clase “alert-danger”. Si el mensaje es positivo, la clase que añadimos es “alert-success”.

La alerta será distinta si hay algún campo no rellenado o si el campo rellenado no cumple una lógica básica. La alerta desaparecerá pasados unos segundos.

Gasto Semanal

Añade tus gastos aqui

Ambos campos son obligatorios

Gasto:
Comida mascotas

Cantidad:
Cantidad en €

Agregar

Listado

Presupuesto: 350€

Restante: 350€

Gasto Semanal

Añade tus gastos aqui

Importe NO válido

Gasto:
Comida mascotas

Cantidad:
holaaaaa

Agregar

Listado

Presupuesto: 350€

Restante: 350€

5. EJECUCIÓN: CREAR GASTO

Cada gasto debe ser un objeto que incluya nombre, cantidad y un id ya que los vamos a poder borrar, por lo que debemos identificarlos (como sugerencia de id, Date.now()).

Cada vez que el usuario crea un gasto y se haya incorporado correctamente al array de gastos, aparecerá una alerta informativa y el formulario se reiniciará.

Gasto Semanal

Añade tus gastos aqui

¡Gasto añadido correctamente!

Gasto:
Comida mascotas

Cantidad:
35

Agregar

Listado

Presupuesto: 600€

Restante: 600€

6. EJECUCIÓN: IMPRIMIR LISTADO DE GASTOS

Cada vez que el usuario incluya un gasto (a través del botón **Agregar**), se debe ver reflejado a la derecha de la pantalla. Cuando se cree un nuevo gasto (a través de elementos de HTML *li*), se le darán las clases 'list-group-item', 'd-flex', 'justify-content-between', align-item-center' (recordamos añadir el id del objeto gasto al elemento creado).

Gasto Semanal

Añade tus gastos aqui

Gasto:
Nombre Gasto

Cantidad:
Cantidad en €

Agregar

Listado

Comida mascotas	35€	Borrar X
Regalo cumple Laura	25€	Borrar X
Compra semanal	152€	Borrar X

Presupuesto: 500€

Restante: 500€

7. EJECUCIÓN: ACTUALIZAR EL PRESUPUESTO RESTANTE

Cada vez que el usuario introduzca un gasto o se borre, el valor del elemento **Restante** debe recalcularse.

Gasto Semanal

Añade tus gastos aquí

Gasto:

Cantidad:

Agregar

Listado

Comida mascotas	35€	Borrar X
Libro	15€	Borrar X

Presupuesto: 175€

Restante: 125€

Cambio de color de *Presupuesto Restante*: Si el usuario tiene un gasto de más del 50% del presupuesto fijado el color del elemento *Restante* cambiará a amarillo y si pasa del 75% cambiará a rojo.

Esto lo gestionaremos a través de las clases:

- Gasto normal: clase alert-succes
- Gasto > 75%: clase alert-danger
- Gasto > 50%: clase alert-warning

Gasto Semanal

Añade tus gastos aquí

Gasto:

Cantidad:

Agregar

Listado

Comida mascotas	35€	Borrar X
Comida mascotas	78€	Borrar X

Presupuesto: 150€

Restante: 37€

Gasto Semanal

Añade tus gastos aquí

Gasto:

Cantidad:

Agregar

Listado

Comida mascotas	50€	Borrar X
Compra semanal	30€	Borrar X

Presupuesto: 150€

Restante: 70€

8. EJECUCIÓN: CONTROL DE EXCESOS DE GASTOS

Si el usuario con un gasto agota o excede el importe restante de presupuesto, se le permitirá añadir el gasto pero se mostrará una alerta que desaparecerá a los segundos indicando que ha agotado el presupuesto y el botón **Agregar** quedará deshabilitado.

Añade tus gastos aqui		Listado	
¡Gasto añadido correctamente!		Comida mascotas	150€
PRESUPUESTO AGOTADO!!		Pago teléfono	95€
Gasto:		Regalo cumple Patricia	35€
Nombre Gasto			
Cantidad:			
Cantidad en €			
Agregar			
		Presupuesto: 250€	
		Restante: -30€	

9. EJECUCIÓN: BORRAR UN GASTO

Borrar un gasto: Cuando el usuario borre un gasto, deberemos eliminarlo y recalculamos el restante, color del elemento....

10. RESUMEN: MÉTODOS Y FUNCIONES (como soporte al desarrollo)

Clase Presupuesto:

- nuevoGasto
- calcularRestante
- eliminarGasto

Clase userInterface:

- imprimirPresupuesto → método inicial para pintar en la pantalla el presupuesto y restante
- imprimirAlerta → cada vez que se pinte un mensaje se hará desde este método
- imprimirGastosListado → encargado de imprimir los gastos en la parte derecha de la pantalla
- limpiarHTML → se encarga de que no haya gastos repetidos en pantalla
- actualizarRestante → pinta el valor actualizado del restante
- comprobarPresupuesto → verifica el % de importe gastado para, utilizando clases, modificar el aspecto de restante. Además comprueba si el importe gastado excede o agota el presupuesto.

Funciones generales:

- preguntarPresupuesto
- añadirGasto → validación de formulario, creación de objeto de gasto...
- eliminarGasto → asociada al botón. Desde ahí se llama al método de la clase.