

# ENTREGA DEL PROYECTO INTERFAZ CRUD DE LA 2ª EVALUACIÓN (calificable)

Se trata de crear una interfaz CRUD aplicando, por una parte, todos los conocimientos HTML+CSS de la primera evaluación (más formularios y tablas), y añadirle además una validación con JS a cada campo introducido.

En primer lugar, y antes de nada, el CRUD deberá estar basado en un diseño relacional entre 2 tablas, que tengan entre sí una relación 1:n.

Dado que en la primera entrega, hubo cierta confusión, pondremos un ejemplo muy muy básico de dicha relación:

EMPLEADO

NOMBRE	APELLIDOS	CODDEPTO
José	Ruiz	1
María	Pérez	2
Elisa	Fernández	1
Rosa	Sanz	3

DEPARTAMENTO

CODIGO	DEPTO
1	Administración
2	Finanzas
3	Comercial

En este ejemplo, la primera tabla EMPLEADO, contiene una clave foránea correspondiente a una segunda tabla DEPARTAMENTO, con la que tiene una relación de “muchos a uno”. Esto es, puede haber más de un empleado que pertenezca a un mismo departamento.

(Esto no tiene por qué verse desde el front, de hecho, está más allá del back, en el diseño de la BD, la capa más interna de cualquier aplicación, pero es importante tenerlo en mente, para lo que sigue)

Para ello, deberá incluirse:

---

## MAQUETACIÓN (5 PUNTOS)

- 4 archivos html que corresponderán a:
  - **login.html**: solo se introducirá usuario y contraseña (esto, en realidad, se almacenaría en la BD en una tabla aparte, pero eso no nos interesa ahora, vamos a centrarnos en lo que sí)
  - **listado.html**: se mostrarán todos los datos de las dos tablas relacionales enlazadas por la relación 1:n, obviando las claves primaria-secundaria, esto es, en el ejemplo : José - Ruiz - Administración, etc, organizadas en filas y columnas (una tabla html, que por razones de responsividad, se podrán convertir en cards)

- **insercion.html:** la introducción de todos los datos. Se deberá incluir, como mínimo:
  - tres campos tipo “text” (ver validaciones más abajo)
  - un campo numérico (lo puedes hacer tipo “text” y luego validarlo para que sea numérico)
  - un campo correo electrónico
  - un campo select que represente la relación 1:n (se despliegan todas las filas posibles de la tabla pequeña, y se debe seleccionar una. Se mostrará el nombre, pero internamente, se tomará el value de la clave, no el string del nombre, en el ejemplo, si se elige “Administración” el value deberá ser 1)
  - un campo radio button (se podrá elegir entre 2 o tres opciones excluyentes)
  - una casilla de verificación (marcar si deseas recibir info o no, si eres mayor de edad o no, si ...cualquier otra cosa, dependiendo del supuesto de BD)
- **modificacion.html:** los mismos campos que en la inserción, pero con contenido supuesto

- 1 solo archivo **estilo.css** , o uno por cada html, incluidos **en una subcarpeta css**

Todo el interfaz deberá tener una estética armoniosa: favicon, colores, tipografías, botones, iconos.... y seguir una tendencia de diseño: brutalista, bauhaus, flat design, neumorfica, glassmorfica, dark....

Sobre todo : **EL INTERFAZ DEBE SER FULL RESPONSIVE**

**IMPORTANTE:** En principio, no es necesario que los cuatro archivos estén encadenados, se podrán abrir independientes unos de otros, por lo que se pide que lo habilites como página web en github, pero subas las URLs de las 4 páginas por separado

**Los mensajes de error deberán integrados en la interfaz de forma armoniosa**

## VALIDACIÓN (5 PUNTOS)

Deberán incluirse validaciones de formularios, nuevamente se da a elegir entre tenerlas en un solo archivo js o en uno por formulario (ten en cuenta que inserción y modificación responden a las mismas validaciones, por lo que, como mucho, deberán ser dos js) **en una subcarpeta js**

Las validaciones deberán ser las siguientes (LEELAS CON ATENCIÓN):

- **login.html:**
  - usuario: deberá no estar vacío y tener una longitud de entre 8 y 12 caracteres
  - contraseña: deberá no estar vacía, tener entre 8 y 15 caracteres, incluir al menos una mayúscula, una minúscula y un dígito
- **listado.html:** no tiene validación, pero sí una ventana de confirmación al pulsar el icono o botón de eliminar/borrar en cada una de las líneas, que diga “Está seguro de

querer eliminar este (cliente, o empleado, o lo que sea...)?” y tenga dos botones: eliminar y cancelar. Es tremendamente sencillo, como un alert, pero trae botones: [https://www.w3schools.com/jsref/met\\_win\\_confirm.asp](https://www.w3schools.com/jsref/met_win_confirm.asp)

- **insercion.html:** Se deberá incluir, como mínimo:
  - un campo text del que se confirme solo que no está vacío
  - otro campo text que deba tener (además) una longitud mínima en caracteres
  - un tercer campo text que responda a un formato (p.ej. Dni, un teléfono, un código especial de lo que sea con, por ejemplo 3 dígitos y 2 caracteres, según el supuesto, tú mismx)
  - un campo numérico (lo puedes hacer tipo “text” y luego validarlo para que sea numérico), con un valor mínimo y un máximo (por ejemplo: años/meses/dias de carnet/suscripción/alquiler/... o cualquier otro supuesto: número de habitación con 3 dígitos....)
  - un campo correo electrónico. Validado
  - un campo select que represente la relación 1:n : deberá tener la opción inválida “elige uno” y se comprobará que no está esa seleccionada. (Recuérdese que se mostrará el nombre significativo de la opción, pero internamente, se tomará el value de la clave, no el string del nombre, en el ejemplo, si se elige “Administración” el value deberá ser 1)
  - un campo radio button (se podrá elegir entre 2 o tres opciones excluyentes). Valida que se elige una, y **por defecto no habrá ninguna marcada**
  - una casilla de verificación (marcar si deseas recibir info o no, si eres mayor de edad o no, si ...cualquier otra cosa, dependiendo del supuesto de BD). Si no estuviera marcada, debe dar error, y no estará marcada por defecto
- **modificacion.html:** los mismos campos que en la inserción, pero con value inicial supuesto. Al poder modificarse por el usuario, deberán ser validados exactamente igual

**SI EL EJERCICIO NO SE ENTREGA A TIEMPO O NO SE PUEDE ABRIR EN EL NAVEGADOR CORRECTAMENTE, NO SERÁ CORREGIDO**

**RÚBRICA DE CORRECCIÓN:** Repasa con cuidado y asegúrate de cumplir con los requerimientos

**MAQUETACIÓN:** Se empieza a corregir con 5 puntos, pero:

- Por cada fallo de responsividad: se quitará un punto
- Por cada fallo en la visualización de mensajes de error y/o confirmación: se quitará medio punto
- Por cada fallo de maquetación (contraste, espaciado, descuadre, tipografía): se quitará medio punto

**VALIDACIÓN:** Se empezará a corregir con 5 puntos, pero:

- Por cada fallo de validación completo (no valida/confirmar nada): se quitará un punto
- Por cada fallo de validación incompleto (valida parcialmente): se quitará medio punto