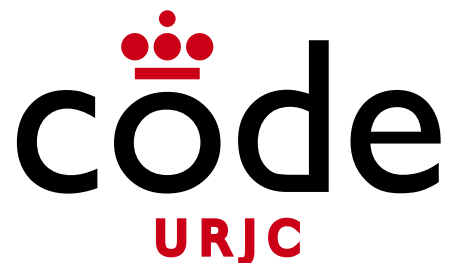


Fundamentos de la Web

# Colección de pruebas de evaluación



©2023

Micael Gallego, Nicolás Rodríguez, Iván Chicano, Michel Maes

Algunos derechos reservados

Este documento se distribuye bajo la licencia  
“Atribución-CompartirIgual 4.0 Internacional”  
de Creative Commons Disponible en  
<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

# Índice de Pruebas de Evaluación

## Prácticas

- Práctica 1 - Convocatoria Ordinaria 2022
- Práctica 2 - Convocatoria Ordinaria 2022

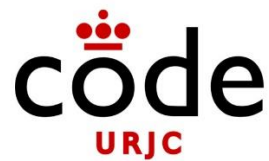
# Índice de Pruebas de Evaluación

## Exámenes

- Exámen Final - Convocatoria Ordinaria 2022
- Exámen Final - Convocatoria Extraordinaria 2023



Universidad  
Rey Juan Carlos



**Fundamentos de la Web**

**2022-2023**

**Enunciado de la práctica I**

# Enunciado

El objetivo de la práctica consiste en implementar una página web en la que se puedan dar de alta, listar, modificar y borrar elementos.

## Información de la web

El tipo de elementos y sus atributos se deja a elección de cada uno de los grupos. A modo de ejemplo, se indican algunos ejemplos de páginas web que se podrían implementar:

- Gestor de notas: Cada elemento será una nota
- Menú de un restaurante: Cada elemento será un plato o bebida del menú
- Tienda: Cada elemento será un producto.
- Periódico: Cada elemento será una noticia.
- Liga de fútbol: Cada elemento será un equipo.

Cada elemento tendrá como atributos un título y una descripción. Adicionalmente se pueden añadir más atributos dependiendo de la naturaleza del elemento (por ejemplo la fecha de creación, precio, categoría, etc.).

Por otro lado, los elementos deberán tener subelementos asociados. En el caso de los ejemplos previos:

- Elemento del gestor de notas: Nota, Subelemento: Cada uno de los items de la nota (por ejemplo una lista de la compra o un lista de cosas por hacer).
- Elemento del menú: Plato, Subelemento: Ingrediente del plato.
- Elemento de la tienda: Producto, Subelemento: Característica técnica del producto.
- Elemento del periódico: Noticia, Subelemento: Comentario.
- Elemento de la liga de fútbol: Equipo, Subelemento: Jugador.

Para simplificar la implementación, un subelemento sólo estará representado por un texto descriptivo.

Cada grupo de prácticas tendrá que elegir el tipo de web que quiere implementar, especificar los atributos de sus elementos y de qué tipo son sus subelementos.

## Navegación por la página web

La página web será una única página web (un único fichero HTML) que irá cambiando sus elementos en función de los links que vaya pulsando el usuario.

- Inicialmente la página mostrará un listado con el nombre de cada uno de los elementos que tenga la página. Al final del listado aparecerá un link "Nuevo elemento".
- Si no hay elementos debe aparecer el texto "Sin elementos".
- Si el usuario pulsa sobre el nombre de un elemento, el listado desaparecerá y se mostrará toda la información de ese elemento.
- Cuando se muestre la información del elemento, aparecerán los siguientes enlaces:
  - Enlace para volver a mostrar la lista de elementos.
  - Enlace para borrar el elemento.
  - Enlace para modificar el elemento.
- Cuando se pulse el enlace para borrar el elemento, deberá aparecer un cuadro de diálogo preguntando al usuario si está seguro de que se borre.
- Cuando se pulse el enlace para modificar el elemento:

- Los atributos de ese elemento deberán convertirse en un formulario, de forma que el usuario pueda cambiar cualquier atributo del elemento. Si se pulsa el enlace “Guardar”, se guardarán los cambios. Si se pulsa el botón “Cancelar”, se volverá a mostrar la información original del elemento.
- Los subelementos también deberán convertirse en campos de texto para que puedan modificarse. Cada uno de los campos de texto de los subelementos tendrá asociado un link “Borrar” que eliminará ese subelemento (previa confirmación con un cuadro de diálogo).
- Para crear subelementos habrá un link “Añadir elemento”, que al ser pulsado añadirá un nuevo campo de texto al formulario.
- Si el usuario está viendo la lista de elementos y pulsa en el link “Crear elemento”, aparecerá un formulario vacío que le permitirá crear un nuevo elemento similar al utilizado para modificar un elemento.

### Características técnicas de la aplicación

La aplicación web se deberá implementar cumpliendo con los siguientes requisitos técnicos:

- La página web se implementará con HTML, CSS y JavaScript en el lado del cliente.
- Los estilos CSS se definirán en uno o varios ficheros .css
- El código JavaScript se definirá en uno o varios ficheros .js
- Se usará una plantilla Bootstrap gratuita disponible en Internet.
- Al cargar la página web deberán cargarse de forma automática datos de ejemplo que sean razonables. Por ejemplo, si la página es un menú de un restaurante, deberán estar cargados platos de un restaurante.
- La página web no tendrá parte servidor. Esto tiene varias implicaciones:
  - La información que introduzca el usuario sólo la podrá consultar él
  - Al recargar la página se destruirá la información insertada o modificada y se volverá a carga la información por defecto.

En la práctica 2 de la asignatura se añadirá un servidor (backend) a la aplicación web, lo que evitará estas limitaciones.

- Para el desarrollo colaborativo de la práctica se usará un repositorio público creado en la plataforma GitHub.

El no cumplir con algún punto del apartado de “[Características técnicas de la aplicación](#)” es motivo de suspenso.

Para facilitar el desarrollo de la práctica se recomienda que se diseñen las diferentes pantallas usando datos de prueba en HTML y CSS (sin JavaScript). De esa forma, es más sencillo convertir el HTML con datos de ejemplo en plantillas con las que implementar la página web.

## Formato y fecha de entrega

La entrega se realizará a través del Aula Virtual de la asignatura en una tarea habilitada al efecto. Se debe entregar el proyecto en un único fichero zip.

La práctica deberá defenderse en clase los días 29 y 30 de noviembre de 2022 (fecha tentativa) de manera presencial.

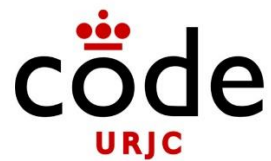
La práctica deberá implementarse en grupos de 2 o 3 personas.

NOTA: Si dos prácticas se parecen “demasiado” se asumirá que han sido realizadas en conjunto y serán penalizadas en la calificación, pudiendo llegar al suspenso. Se revisará GitHub en caso de duda.





Universidad  
Rey Juan Carlos



**Fundamentos de la Web**

**2022-2023**

**Enunciado de la práctica II**

## Enunciado

El objetivo de la práctica consiste en implementar una página web en la que se puedan dar de alta, listar, modificar y borrar elementos. Esta práctica II es la continuación de la práctica I.

### Información de la web

Esta práctica II es la continuación de la práctica I, con los respectivos elementos y subelementos.

### Navegación por la página web

La página web serán varias páginas web (puede haber varios ficheros HTML, aparte de plantillas) que irán cambiando sus elementos en función de los links o botones que vaya pulsando el usuario. El resto de la funcionalidad que se pide es la misma que en el apartado “[Navegación por la página web](#)”, en la práctica I. Se pide añadir la siguiente funcionalidad:

1. Los elementos que se muestran están almacenados en un fichero “`{{element}}Service.js`”, donde `{{element}}` es el nombre de nuestro elemento. En la página principal se pide mostrar todos los elementos.
2. Cada vez que añada un elemento, se añade en el “`{{element}}Service.js`”. El id es auto incremental, es decir, no es un parámetro a insertar.
3. Cada vez que se actualice un elemento, se tiene que actualizar en el “`{{element}}Service.js`”. Para ello es necesario utilizar el id del elemento. **No se permite borrar un elemento y luego crear uno “nuevo actualizado”**.
4. Cada vez se borre un elemento, se tiene que borrar en el “`{{element}}Service.js`”, a través de su id. Dado que el id es auto incremental en la creación, ese id tiene que quedarse vacío y no volver a usarse.
5. Consultar un elemento en detalle a través de su id.
6. Los puntos 2, 3 y 4 se aplican también sobre los subelementos.
7. En el punto 1, se pide además que se muestren los elementos de 5 en 5. Inicialmente puede haber más de 5 elementos, pero solo se han de mostrar los 5 primeros. Es necesario cargar más elementos a través de un botón “cargar más”. Se pide utilizar AJAX. **Este punto es obligatorio.**

### Características técnicas de la aplicación

La aplicación web se deberá implementar cumpliendo con los siguientes requisitos técnicos. Las mismas que en el apartado de “[Características técnicas de la aplicación](#)” salvo:

- La página web tendrá parte servidor. Esto tiene varias implicaciones:
  - Hay que usar Node.js para para la parte servidor.
  - Al recargar la aplicación de node se destruirá la información insertada o modificada y se volverá a carga la información por defecto. Mientras la aplicación esté lanzada, la información tiene que ser “persistente”.
- **La información se almacena en un mapa**

El no cumplir con algún punto del apartado de “[Características técnicas de la aplicación](#)” es motivo de suspenso.

## **Formato y fecha de entrega**

La entrega se realizará a través del Aula Virtual de la asignatura en una tarea habilitada al efecto. Se debe entregar el proyecto en un único fichero zip.

La práctica deberá defenderse en el aula (Laboratorios Polivalentes II - Aula Informática 106) de manera presencial el día 9 de enero de 2023, desde las 17:00 en adelante. Más adelante habrá una planificación

La práctica deberá implementarse en grupos de 2 o 3 personas. Los mismos que en la primera práctica. Si hay algún cambio, es necesario notificarlo al profesor.

NOTA: Si dos prácticas se parecen “demasiado” se asumirá que han sido realizadas en conjunto y serán penalizadas en la calificación, pudiendo llegar al suspenso. Se revisará GitHub en caso de duda.

### **Ejercicio (8 puntos)**

Se pide implementar una página web para gestionar elementos con las siguientes características:

- Al cargar la página web deberá aparecer un listado de elementos.
- Por cada elemento, se mostrará su nombre y su descripción.
- Aparecerá un enlace “Crear elemento” que al ser pulsado reemplazará la lista de elementos por un formulario con el que crear un nuevo elemento.
- Este formulario tendrá los campos para incluir el nombre y la descripción y un botón de “Crear elemento”
- Al pulsar el botón de creación, se añadirá el elemento a la lista de elementos y se volverá a sustituir el formulario por el listado de elementos (que deberá incluir al nuevo elemento).
- No hay que implementar funcionalidad para borrar o para modificar un elemento.

### **Requisitos técnicos de la aplicación:**

- La aplicación se ejecutará completamente en el navegador web. Es decir, no se ejecutará en Node.js.
- Los elementos se podrán almacenar en un array o un mapa.
- La estructura de datos deberá llamarse “elementos”
- Para la manipulación de la página se podrá usar la librería DOM o jQuery
- Para gestionar los elementos se deberá crear la clase Elemento con los atributos:
  - Nombre
  - Descripción
- Se pide crear, al menos, las siguientes funciones con las siguientes responsabilidades:
  - mostrarElementos: Visualiza los elementos de la lista en la página web (modificando el DOM)
  - mostrarFormulario: Muestra el formulario de creación de elemento.
  - crearElemento: Obtiene los datos del formulario, crea un elemento y lo añade a la lista de elementos.
- Se debe entregar un fichero HTML y otro JavaScript.
- Al cargar la página debe haber 3 elementos por defecto.

**Otras consideraciones para la realización del examen:**

- Para la realización de la aplicación, se podrán consultar únicamente las presentaciones en formato PDF publicadas por los profesores. No se podrá consultar ninguna página web adicional ni el código fuente de ejemplos o ejercicios resueltos.
- Sólo se podrá tener abierto en el PC del aula el Visual Studio Code, el navegador con la aplicación ejecutándose y el navegador con el aula virtual para descargar y visualizar las presentaciones de la asignatura.
- Para entregar el examen, se deberán comprimir los ficheros HTML y JS en un .zip que será subido en la tarea “examen” del aula virtual.
- Para la realización del ejercicio práctico se dispondrá de 1 hora.

### **Ejercicio (8 puntos)**

Se pide implementar una página web para gestionar subelementos con las siguientes características:

- Al cargar la página web deberá aparecer un listado de 3 elementos.
- Por cada elemento, se mostrará su nombre y será un enlace.
- Al pulsar este enlace, reemplazará la lista de elementos por la lista de subelementos de ese elemento concreto.
- Habrá un botón o enlace llamado “Crear subelemento” con el que crear nuevos subelementos a través de un formulario.
- Este formulario tendrá los campos para incluir el nombre y la descripción y un botón de “Añadir subelemento”
- Al pulsar el botón de creación, se añadirá el subelemento a la lista de subelementos y se volverá a sustituir el formulario por el listado de subelementos (que deberá incluir al nuevo subelemento).
- No hay que implementar funcionalidad para borrar o para modificar un elemento o subelementos.

### **Requisitos técnicos de la aplicación:**

- La aplicación se ejecutará completamente en el navegador web. Es decir, no se ejecutará en Node.js.
- Los subelementos se podrán almacenar en un array o un mapa.
- Las estructuras de datos deberán llamarse “subelementos”
- Para la manipulación de la página se podrá usar la librería DOM o jQuery
- Para gestionar los elementos se deberá crear la clase SubElemento con los atributos:
  - Nombre
  - Descripción
- Se pide crear, al menos, las siguientes funciones con las siguientes responsabilidades:
  - mostrarSubElementos: Visualiza los subelementos de un elemento en la página web (modificando el DOM)
  - mostrarFormulario: Muestra el formulario de creación de un subelemento.
  - crearSubElemento: Obtiene los datos del formulario, crea un subelemento y lo añade a la lista de subelementos.
- Se debe entregar un fichero HTML y otro JavaScript.
- Al cargar la página debe crear debe haber 3 subelementos por defecto en cada elemento.

**Otras consideraciones para la realización del examen:**

- Para la realización de la aplicación, se podrán consultar únicamente las presentaciones en formato PDF publicadas por los profesores. No se podrá consultar ninguna página web adicional ni el código fuente de ejemplos o ejercicios resueltos.
- Sólo se podrá tener abierto en el PC del aula el Visual Studio Code, el navegador con la aplicación ejecutándose y el navegador con el aula virtual para descargar y visualizar las presentaciones de la asignatura.
- Para entregar el examen, se deberán comprimir los ficheros HTML y JS en un .zip que será subido en la tarea “examen” del aula virtual.
- Para la realización del ejercicio práctico se dispondrá de 1 hora.