

# PEC 1

## Introducción al desarrollo frontend...

UOC

Universitat Oberta  
de Catalunya

### Información relevante:

- Fecha límite de entrega: 4 de octubre.
- Peso en la nota de FC: 12.5%.



## Contenido

Información docente	3
Presentación	3
Objetivos	3
Enunciado	4
Tarea 1 – Configuración del entorno desarrollo	4
Ejercicio 1 – Preguntas teóricas (2 puntos)	5
Ejercicio 2 – Prácticas guiadas (8 puntos)	7
Formato y fecha de entrega	10

# Información docente

## Presentación

Esta Práctica se centra en conocer las tecnologías Web actuales, así como en aprender a configurar un entorno de trabajo para comenzar a desarrollar aplicaciones web. Además, se repasarán conceptos de programación aplicados a JavaScript utilizando la versión ES5 (2009) del estándar de JavaScript.

## Objetivos

Los objetivos que se desean lograr con el desarrollo de esta PEC son:

- **Conozcas las tecnologías web actuales**, y dónde se ubica cada una de ellas en el stack tecnológico.
- Instales un **IDE de desarrollo (Visual Studio Code)** así como la **plataforma de desarrollo web (Node.js)**.
- **Repases conceptos de JavaScript (ES5)** a través de prácticas guiadas.
- Te familiarices con la **dinámica de la asignatura**: metodología, tipología de enunciados, nomenclaturas, etc.

# Enunciado

Esta PEC contiene 1 tarea (actividad no evaluable) y 2 ejercicios (sí evaluables). Debes entregar tu solución de los 2 ejercicios evaluables (ver el último apartado).



Debido a que las actividades están encadenadas (i.e. para hacer una se debe haber comprendido la anterior), **es altamente recomendable hacer las tareas y ejercicios en el orden en que aparecen en este enunciado.**

## Tarea 1 – Configuración del entorno desarrollo

En esta primera PEC se va a configurar un entorno de desarrollo básico para comenzar a trabajar con JavaScript.

- **Visual Studio Code**
- **Node.js y npm.**

Para realizar esta configuración, se facilita el siguiente recurso multimedia que puedes seguir si tienes alguna duda con la instalación de Visual Studio Code o node.js:

- [https://learning.oreilly.com/videos/the-complete-node-js/9781789955071/9781789955071-video2\\_2](https://learning.oreilly.com/videos/the-complete-node-js/9781789955071/9781789955071-video2_2)

En los videos de las prácticas guiadas del ejercicio 2, el autor usa algunas extensiones de Visual Studio como:

- Auto-rename-tag
- Bracket pair colorizer 2
- Javascript (ES6) Code Snippets
- Live Server

Todas son opcionales, pero esta última de Live Server os la recomiendo instalar ya que os permitirá ir viendo el resultado de vuestro Html, Css, Javascript según vayáis codificando y guardando sin tener que recargar el navegador

El autor de los videos del ejercicio 2, también se ayuda mucho de Emmet, esta extensión ya forma parte de Visual Studio Code y no es necesario instalar nada, podéis consultar cómo funciona en Visual Studio Code en el siguiente enlace a la documentación y la veréis en uso en las prácticas guiadas del ejercicio 2, es bastante útil y ahorra bastante tiempo.

<https://code.visualstudio.com/docs/editor/emmet>

## Ejercicio 1 – Preguntas teóricas (2 puntos)

Antes de continuar debes:

Haber leído los recursos teóricos disponibles en la PEC.

1. La aparición de HTML5/CSS3/JS ha supuesto el nacimiento del desarrollo front-end moderno. (0.75 puntos)
  - ¿Cuál es la ventaja del uso de etiquetas semánticas? Nombra y explica al menos 3 de ellas.
  - Cita al menos 3 APIs HTML5 y explica brevemente su funcionalidad.
  - Cita qué opción ofrece CSS3 para conseguir que se apliquen diferentes estilos CSS sobre el mismo elemento en su visualización en diferentes dispositivos (diferentes tamaños de pantalla)
  - Cita al menos 4 de las características principales de TypeScript (importante superset de JavaScript que trataremos en el siguiente capítulo).

2. El lenguaje CSS es muy rígido, poco práctico y ordenado a la hora de programar. Para evitar este problema se han creado los preprocesadores CSS, que ofrecen evidentes ventajas (0.5 puntos)
  - Cita al menos 2 de estos preprocesadores
  - Cita al menos 4 ventajas que ofrecen estos preprocesadores
  - Explica brevemente en qué consisten los sourcemaps
  - Explica qué es un transpilador
3. El flujo de trabajo profesional en front-end hace indispensable el uso de herramientas como controles de versiones o build systems (automatizadores como Gulp/Grunt) – (0.75 puntos).
  - Cita al menos dos sistemas de control de versiones
  - Cita y explica al menos 3 comandos de Git
  - Cita y explica brevemente las características más definitorias de dos build systems

## Ejercicio 2 – Prácticas guiadas (8 puntos)



Estas prácticas son guiadas mediante la visualización de videos, pero se valora positivamente cuando se hagan mejoras sobre el ejercicio propuesto.

### 1. Validador de formularios (2 puntos).

Siguiendo lo que se realiza en el video ejemplo, debéis:

- Construir un formulario HTML con varios campos a completar (Username, Email, Password, Confirm Password), sobre estos campos debéis realizar diferentes validaciones.
- A parte de las validaciones que se realizan en el video, añadir la comprobación de que la contraseña esté formada por: números, letras mayúsculas, letras minúsculas y un símbolo.

Video ejemplo para seguir:

[https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video2\\_1](https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video2_1)

### 2. Reserva de asientos de cine (2 puntos).

Siguiendo lo que se realiza en el video ejemplo, debéis:

Realizar una pequeña aplicación para la reserva de asientos de películas, se utilizará almacenamiento local del navegador (es similar a utilizar un fichero de texto, pero almacenado en el navegador).

Video ejemplo para seguir:

[https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video3\\_1](https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video3_1)

### 3. Calculadora de tipos de cambio de moneda (2 puntos)

Siguiendo lo que se realiza en el video ejemplo, debéis:

Hacer uso de código JavaScript para llamar a una API de terceros. Esta Api nos servirá para que facilitándole una moneda base, nos devuelva el cambio al que cotiza esa moneda base respecto a otras muchas.

El objetivo de esta parte práctica es realizar peticiones asíncronas con fetch a un API de terceros).

En el video se usa la versión 4 de la API de [api.exchangerate-api.com](https://api.exchangerate-api.com/), actualmente van por la Versión 6, no obstante, la versión 4 todavía funciona y es válida, podéis hacerlo con cualquiera de estas versiones del API o incluso con otra API que ofrezca este servicio.

Video ejemplo para seguir:

[https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video5\\_1](https://learning.oreilly.com/videos/20-web-projects/9781800563049/9781800563049-video5_1)

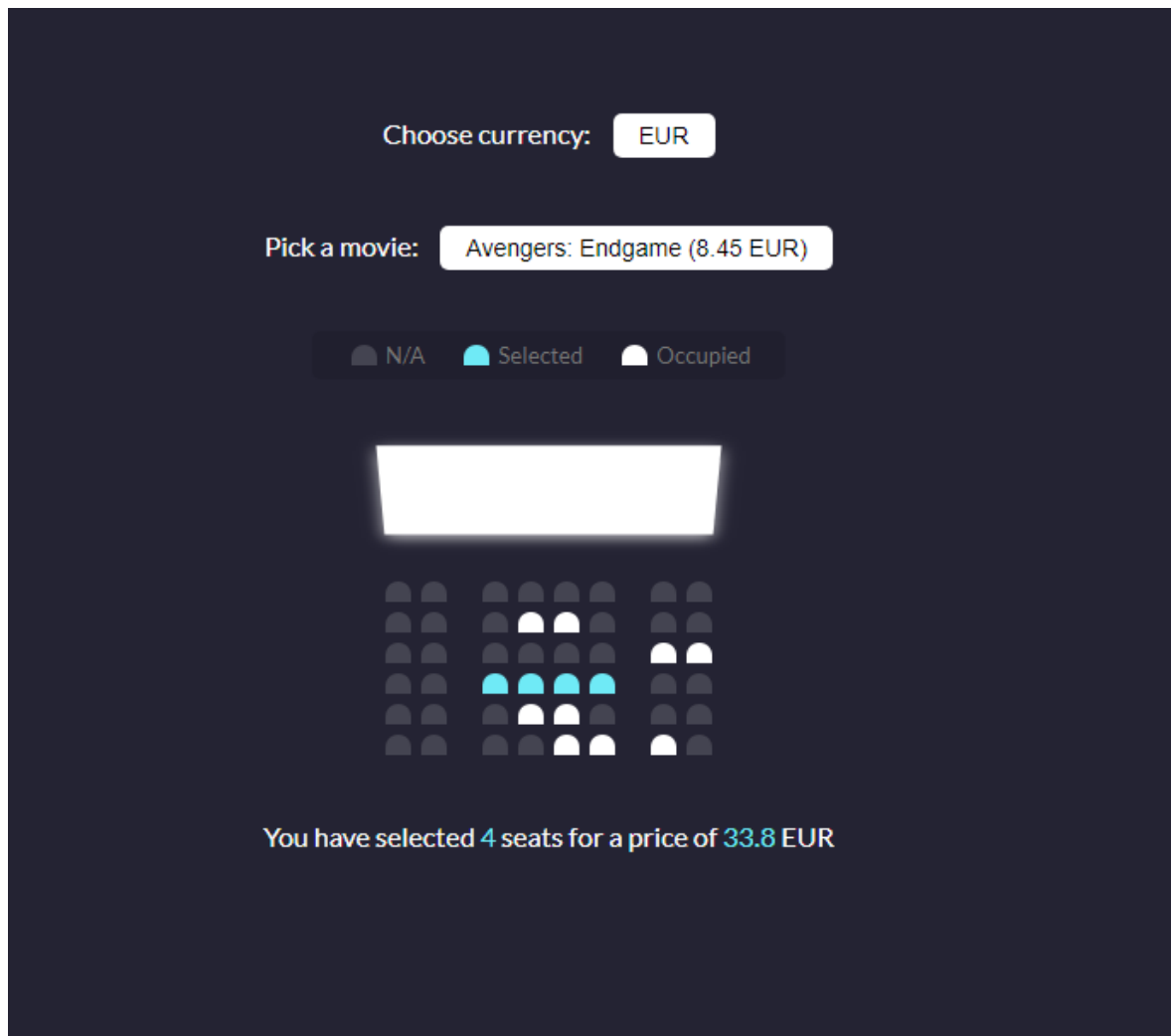
### 4. Integración de los cambios de moneda en la reserva de asientos de cine (2 puntos).

Una vez realizada la parte 3 de este ejercicio (Calculadora de tipos de cambio de moneda) y la parte 2 (Reserva de asientos de cine)

Integrar estas dos partes de forma que la aplicación realizada para reserva de asientos de cine permita seleccionar la moneda en la que trabajara la aplicación y se ofrezcan tanto los precios de las películas, como los precios finales en la moneda seleccionada.

Por ejemplo, sería algo como esto:





Debéis incluir, un desplegable (o similar) que permita seleccionar la moneda y al cambiar la moneda seleccionada debéis hacer:

- Que cambie el importe de la entrada, si antes la entrada de Avengers: Endgame, eran 10 dólares, ahora al seleccionar la moneda EUR se aplicara la conversión. En el ejemplo he supuesto que 10 Dólares equivalen a 8.45 EUR, esta conversión debe realizarla online el programa consultando el API usada en el ejercicio 3
- También debe cambiar el importe total de las entradas. Si antes 4 entradas tenían un precio de 40 dólares, ahora debe mostrar el equivalente en la moneda seleccionada, en el ejemplo 33.8 EUR.

# Formato y fecha de entrega

Tienes que entregar un fichero \*.zip, cuyo nombre tiene que seguir este patrón: loginUOC\_PEC1.zip. Por ejemplo: dgarciaso\_PEC1.zip. Este fichero comprimido tiene que incluir los siguientes elementos:

- Un documento de texto S01\_PEC1\_Solución\_Teoría con el enunciado y las respuestas a las preguntas de los tres apartados del ejercicio 1
- Una carpeta PEC1\_Ej2\_1 con el código completado siguiendo las peticiones y especificaciones del Ejercicio 2.1.
- Una carpeta PEC1\_Ej2\_2 con el código completado siguiendo las peticiones y especificaciones del Ejercicio 2.2.
- Una carpeta PEC1\_Ej2\_3 con el código completado siguiendo las peticiones y especificaciones del Ejercicio 2.3.
- Una carpeta PEC1\_Ej2\_4 con el código completado siguiendo las peticiones y especificaciones del Ejercicio 2.4.

El último día para entregar esta PEC es el **4 de octubre de 2020** hasta las **23:59**. Cualquier PEC entregada más tarde será considerada como no presentada.