

# Proyecto Final: Cuadro de Mandos Interactivo con HTML, CSS, JavaScript y Librerías de Datos y Visualización

## Descripción del Proyecto:

En este proyecto final, los estudiantes tendrán la oportunidad de aplicar sus conocimientos en desarrollo web utilizando tecnologías fundamentales como HTML, CSS y JavaScript, además de utilizar librerías especializadas en manipulación de datos y visualización como **dataframe-js** y **plotly-js**. El objetivo principal es crear un cuadro de mandos interactivo que presente información relevante de manera efectiva y atractiva.

Tienes disponible en Google classroom y en el aula virtual el siguiente link:

<https://www.loom.com/share/2de4e21ec68c4501a0473e8085283040?sid=027ad5a5-eb8c-47af-97c5-ad6154c47f14> con una demo que muestra que funcionalidades ha de tener nuestra página web.

A modo de resumen, nuestra página parte del formulario de la práctica anterior, donde el usuario ha de introducir sus datos de registro y ha de ser redirigido a la página principal del **dashboard**.

Inicialmente, en el dashboard tendríamos que tener un menú en la cabecera, un **dragable**, una sección **About-Me**, en la que hablamos de nosotros, una foto nuestra (o inventada) y enlaces de interés a nuestras redes sociales (o inventadas) y nuestro GitHub.



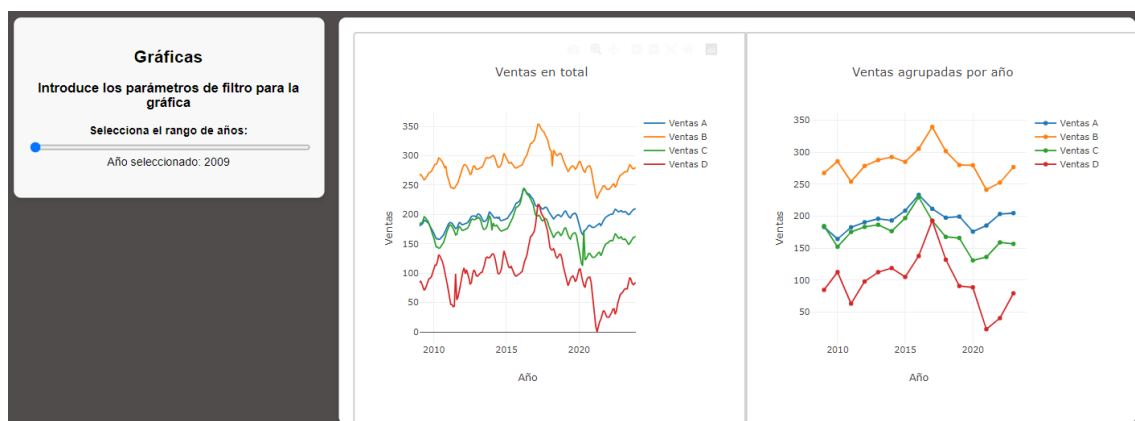
Una vez que subamos el fichero con nuestros datos, tanto arrastrando como haciendo click, la página web se tiene que actualizar mostrándonos dos nuevas secciones: gráficas y tablas. Estas dos secciones aparecerán tanto en contenido de la página, como en el menú de la cabecera y en el pie de página.

Además, debajo del draggable aparecerán 4 elementos de navegación para navegar a la introducción, a las gráficas, a las tablas, a la información personal (*About Me*) y para salir de la sesión (volver a la página de inicio).



Una vez subamos el fichero, lo tendremos que preprocesar eliminando datos ausentes, corrigiendo el tipo de datos,... y mostraremos las gráficas y tablas correspondientes.

En la sección de gráficas tendremos lo siguiente:



Tendremos un **slider** en el que seleccionaremos desde que año queremos visualizar los datos. Las gráficas a mostrar serán dos series temporales:

- a) Serie temporal de las ventas de los cuatro productos de cada mes
- b) Serie temporal de las ventas de los cuatro productos agrupada por la media de cada año

Del mismo modo que con las gráficas, en la sección de las tablas tendremos otro **slider** con el que seleccionaremos desde que año queremos mostrar los datos y mostraremos dos tablas: las ventas de cada mes y la media de ventas de cada año:

Tablas

Introduce los parámetros de filtro para la tabla

Selecciona el rango de años:  Año seleccionado: 2009

Tabla total

Tabla de datos

Year	Month	VentasA	VentasB	VentasC	VentasD
2009	Enero	180.05	266.66	182.11	84.15
2009	Febrero	182.47	267.92	185.38	86.12
2009	Marzo	182.61	265.80	185.24	82.47
2009	Abril	185.35	262.09	188.98	76.16
2009	Mayo	190.12	258.99	195.77	70.87
2009	Junio	189.57	260.62	194.60	73.47
2009	Julio	187.77	264.33	191.54	79.52
2009	Agosto	186.48	269.01	189.24	87.20
2009	Septiembre	183.42	271.21	184.29	90.74
2009	Octubre	180.14	271.73	179.01	91.48
2009	Noviembre	176.28	273.37	172.86	94.09
2009	Diciembre	171.86	277.38	165.86	100.66
2010	Enero	167.65	281.99	159.19	108.21
2010	Febrero	161.69	285.37	149.90	113.72
2010	Marzo	159.45	285.24	144.68	113.54

Tabla agrupada

Tabla de datos agrupados

Year	VentasA	VentasB	VentasC	VentasD
2009	183.01	267.43	184.57	84.74
2010	164.29	285.72	152.17	112.65
2011	182.64	253.98	175.38	63.40
2012	190.48	278.51	183.19	97.87
2013	195.88	287.64	186.59	112.51
2014	193.28	292.36	176.52	118.90
2015	208.58	284.92	197.01	105.02
2016	233.28	305.41	229.74	137.69
2017	211.50	339.46	192.75	192.96
2018	197.62	301.61	167.61	132.03
2019	199.39	279.97	165.94	90.86
2020	175.84	279.62	130.87	88.79
2021	185.32	241.30	136.20	23.44
2022	203.43	252.53	159.04	40.67
2023	204.73	272.60	155.66	70.45

Las gráficas se actualizan nada más se mueva el **slider**, al igual que las tablas

# Detalles del proyecto

## 1. Implementación de HTML → 2 puntos

### 1.1. **Cabecera:** Crear elementos de navegación en la cabecera con

- Introducción
- Gráficas
- Tablas
- About Me
- Exit

Cada elemento tiene que redirigirte a la sección correspondiente en la página ***dashboard.html***

### 1.2. **Sección 1: Introducción**

#### 1.2.1 Implementar un ***div*** con el ***draggable***

1.2.2. Implementar un ***div*** contenedor en el que aparezcan los elementos de la navegación entre secciones con una breve descripción de cada sección

### 1.3. **Sección 2: Gráficas**

#### 1.3.1. ***div*** que contenga el título e implemente un ***slider***:

- Fija el valor mínimo a 2009 y el valor máximo a 2023
- Fija el valor por defecto a 2009

#### 1.3.2. ***div*** que contenga los resultados de las gráficas:

- ***Div*** donde se mostrará el resultado de la serie temporal de cada mes
- ***Div*** donde se muestre la serie temporal agrupada por la media del año

### 1.4. **Sección 3: Tablas**

#### 1.4.1. ***div*** que contenga el título e implemente un ***slider***:

- Fija el valor mínimo a 2009 y el valor máximo a 2023
- Fija el valor por defecto a 2009

**1.4.2. *div* que contenga los resultados de las gráficas:**

- ***Div*** donde se mostrará el resultado de la serie temporal de cada mes
- ***Div*** donde se muestre la serie temporal agrupada por la media del año

**1.5. Sección 4: About-Me**

**1.5.1. Título**

**1.5.2. Descripción**

**1.5.3. *Div* donde se colocan los enlaces a las redes sociales**

- Enlace a la red social
- Ícono de la red social

**1.6. Pie de página**

- Elementos de navegación
- Enlaces de redes sociales

## 2. Implementación de JavaScript → 7 pt

Algunas animaciones del estilo las puedes crear directamente en CSS e importarlas con el método **.classList**

### 2.1. Redirección de la página del Login a dashboard

- Cuando los datos introducidos por el usuario corresponden con un registrado, tiene que redirigirte la página a **dashboard.html**

### 2.2. Ocultar al inicio de la página:

- El **div** contenedor de los elementos de navegación en la sección de la introducción
- La sección de la gráfica y de la tabla
- Las opciones de gráfica y tabla en el menú de la cabecera y en el menú del pie de página

### 2.3. Animaciones de elementos de navegación cabecera y pie de página

- Al pasar el ratón, cambiamos el color del fondo
- Al quitar el ratón, devolvemos el color del fondo original

### 2.4. Animaciones del **dragable**

- a) Al pasar el ratón por encima, cambiamos:
  - Color del fondo
  - Color del borde
- b) Al quitar el ratón: devolvemos el color original del fondo y del borde
- c) Cuando un archivo está siendo arrastrado sobre el área del **dragable**, cambiamos:
  - Color del fondo
  - Color del borde
  - Aumentamos tamaño
- d) Cuando el archivo sale del área, recuperamos el color y el tamaño de antes
- e) Procesamos un archivo (ejecutamos el método **processFile()**, definido más adelante) cuando:
  - Es soltado en el área del dragable
  - Hacemos click en el área del **dragable**

## 2.5. Implementación del método **AddEventsOnContainer()**:

implementamos las siguientes animaciones a todos los elementos de navegación del **div** contenedor en la sección de la introducción

- a) Cambio de color del fondo cuando pasamos el ratón por encima y vuelta del color cuando quitamos el ratón
- b) Redirigirte a la sección correspondiente al hacer **click**

## 2.6. Implementación del método **processFile**.

- a) Realizar lectura en navegador
- b) Al cargar el lector:
  - a. Mostrar:
    - Elementos de navegación del **div** contenedor de la sección de la introducción
    - Las secciones de las gráficas y de las tablas
  - b. Ejecutar el método **AddEventsOnContainer()**
  - c. Obtención del **DataFrame**, realizar las siguientes acciones utilizando una promesa
    - Obtención de un dataframe preprocesado ejecutando el método **preprocessDF**
    - Ejecución del método **showPlots()** (definido más adelante)
    - Ejecución del método **showTable()** (definido más adelante)

## 2.7. Implementación del método **preprocessDF**

- a) Coge como parámetro un *dataframe*
- b) Realiza la conversión de cada columna al formato adecuado
- c) Rellena los valores ausentes de cada columna con la media de la misma

## 2.8. Implementación del método **roundDigitsDF**

- a) Coge como parámetro un *dataframe*
- b) Redondea cada columna a dos decimales

## 2.9. Añadir las funcionalidades de los **sliders**

- a) Tiene que aparecer el valor del año seleccionado por el **slider**
- b) Si se cambia el valor del **slider**
  - a. Se actualiza el valor que aparecía antes
  - b. Se filtra el **dataframe**
  - c. Se ejecuta el método **showPlots()** con el slider de la sección de las gráficas y el método **showTable()** con el **slider** de la sección de las tablas

## 2.10. Crear el método **groupedDF**

- Coge como parámetro un dataframe
- Devuelve el dataframe agrupado por el año, por la media en cada columna

## 2.11. Crea el método **getDateFromDF**

- Coge como parámetro un dataframe
- Devuelve un array con las fechas en formato **nativo** de JavaScript (**Date**)

## 2.12. Implementa el método **showPlots()**

Este método será ejecutado cuando se lea el **dataframe** y cuando modifiquemos su **slider**

- a) Filtra el dataframe por año (según el valor del **slider**)
- b) Obtén el array de fechas en formato nativo de JavaScript con el método **getDateFromDF** definido previamente
- c) Obtén el dataframe agrupado por año con la media usando el método **groupedDF** definido previamente
- d) Realiza la gráfica de la serie temporal usando plotly de cada columna con los datos mensuales y con los datos agrupados y haz que el resultado aparezca en la etiqueta HTML correspondiente definida en el apartado anterior



### 2.13. Implementa el método ***showTables()***

- a) Filtra el dataframe por año (según el valor del *slider*)
- b) Redondea cada dataframe a dos decimales usando el método ***roundDigitsDF()***
- c) Obtén el dataframe agrupado por año
- d) Muestra la tabla formateada usando plotly.js de los datos mensuales y los datos agrupados

### 3. Estilo y apariencia → 1pt

Debes crear una interfaz de usuario atractiva utilizando HTML y CSS para diseñar el cuadro de mandos. Asegúrate de que la disposición de los elementos sea lógica y fácil de entender para el usuario. Se valorarán detalles como:

- Elección y contraste de colores
- Forma en la que se disponen los elementos
- Tamaños de las imágenes y los elementos
- Animaciones Adicionales
- ...

### 4. Subir contenido a Github Pages → +1pt

Lograr subir el proyecto a GitHub Pages, de modo que sea accesible la página desde una URL pública

Sube nota hasta el tope máximo:

- 10 si se hace el proyecto desde 0
- 7 si se utiliza las plantillas de HTML y CSS proporcionadas por el profesor