

# TAREA

## ADAT\_Calculadora\_con\_JQuery\_y\_Bootstrap

**Autor:** Marta\_Molina

**Fecha:** 14\_10\_2021

CFGs DAMP

## Índice

1. Introducción.....	3
2. Características.....	3
3. Desarrollo.....	4
3.1. index.html .....	4
3.1.1 <head>...</head> .....	4
3.1.1 <body>...</body> .....	4
3.2. styles.css.....	6
3.3. app.js.....	6
3.3.1 Declaración de variables .....	6
3.3.2 enableSegundoCampo() y disableSegundoCampo().....	7
3.3.3 desactivarBtns() y activarBtns () .....	7
3.3.4 recogerNums () .....	7
3.3.5 operar () .....	8
3.3.6 annadirResultado ().....	8
3.3.7 appenAlHistorial () .....	8
3.3.8 mostrarResultado().....	9
3.3.9 Click listeners .....	9
Bibliografía (Enlaces consultados) .....	11
Índice de figuras.....	11
Índice de tablas.....	11
GLOSARIO.....	12

## 1. Introducción

La aplicación presentada en esta tarea es una calculadora sin botones numéricos, solamente operacionales de: raíz cuadrada, seno, coseno, tangente, cotangente, suma resta, multiplicación, división e igual. La cual presenta dos inputs, activándose el segundo en función de si es preciso su uso o no.

Las tecnologías aplicadas a este proyecto son:

- HTML
- CSS
- JavaScript
- JQuery
- Bootstrap

## 2. Características

La aplicación cuenta con las siguientes características:

FrontEnd	
index.html	Crea los elementos visuales de la app a través de la manipulación del DOM, este documento es el que es leído por el navegador. Enlaza a él la hoja de estilos y los scripts del proyecto. Aplica las clases de Bootstrap.
styles.css	Hoja de estilos ajenos a Bootstrap

**Tabla 1 Front End**

BackEnd	
App.js	Contiene la lógica operacional de la aplicación.

**Tabla 2 BackEnd**

### 3. Desarrollo

#### 3.1. index.html

Documento el cual contiene los metadatos y elementos de la aplicación. Enlaza las hojas de estilo y los scripts.

##### 3.1.1 <head>...</head>

Provee los metadatos del documento: lenguaje, set de caracteres, autor, relación de tamaño con la pantalla en la que es mostrado, enlaza las hojas de estilos de Bootstrap y la creada con CSS, por último da título al documento.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta name="author" content="Marta Molina Aguilera" />

    <!-- Bootstrap CSS -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.2/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-uWxY/CJNBR+1zjPwMfnSnVxwRheevXITnMqoEIEG1LJrdI0G1Vs/9cVSyPYXdcSF"
      crossorigin="anonymous"
    />
    <!-- Hojas de estilos -->
    <link rel="stylesheet" href="./styles/styles.css" />
    <title>Calculadora-MartaMolina</title>
  </head>
```

Captura 1 HTML-head

##### 3.1.1 <body>...</body>

Contiene el contenido del documento, crea lo que son los elementos que se mostrarán al cargar la página.

```
<body class="bg-primary bg-gradient">
  <!--Cabecera-->
  <div class="container-fluid d-flex justify-content-center flex-wrap p-4 mb-4" id="header">
    <h1 class="col-9 text-center p-2 text-light">
      | Calculadora con JQuery y Bootstrap
    </h1>
    <h2 class="col-9 text-center text-light opacity-75">
      | Marta Molina Aguilera
    </h2>
  </div>
```

Captura 2 HTML Body 1

```
<div class="
  containerfluid
  d-flex
  flex-column
  justify-content-center
  align-items-center
  " id="container">
  <div class="col-lg-2 p-4 bg-light rounded-3" id="calculadora">
    <form class="col-12">
      <div class="form-group m-2" id="primer-num">
        <label for="formGroupExampleInput" class="text-dark fw-bold mb-2">Primer número</label>
        <input type="text" class="form-control border-info" id="formGroupExampleInput" placeholder="Primer número " />
      </div>
      <div class="form-group m-2 opacity-50" id="segundo-num">
        <label for="formGroupExampleInput2" class="text-dark fw-bold mb-2">Segundo número</label>
        <input type="text" class="form-control border-info" id="formGroupExampleInput2" placeholder="Segundo número"
          disabled="true" />
      </div>
    </form>

    <div class="
      pt-3
      pb-3
      d-flex
      m-2
      flex-row flex-wrap
      justify-content-between
      align-items-center
      " id="grupo_botones">
      <div class="col-12 mb-2 d-flex justify-content-between">
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnSumar">
          +
        </button>
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnRestar">
          -
        </button>
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnMultiplicar">
          x
        </button>
      </div>
      <div class="col-12 mb-2 d-flex justify-content-between">
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnDividir">
          ÷
        </button>
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnRaiz">
          √
        </button>
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnSeno">
          Sen
        </button>
      </div>
      <div class="col-12 mb-2 d-flex justify-content-between">
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnCoseno">
          Cos
        </button>
```

Captura 3 HTML Body 2

```
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnTan">
          Tan
        </button>
        <button type="button" class="btn btn-primary col-3 btnOperacion" id="btnCotangente">
          Cot
        </button>
      </div>
      <button type="button" class="btn btn-primary col-12" id="btnIgual">
        =
      </button>
    </div>
  </div>
  <div id="resultado" class="mt-2">
    <p id="muchotexto" class="p-4 h2 text-light text-center">Resultado</p>
    <h2 class="h4 text-light text-center opacity-50">Historial</h2>
    <p id="historial" class="small text-center text-light opacity-50"></p>
  </div>
</div>
<!--jQuery script-->
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<!--Script JQuery y JS-->
<script src="js/app.js"></script>
</body>
</html>
```

Captura 4 HTML Body 3

### 3.2. styles.css

Aporta el estilo estético a muy pocos elementos de la tarea, ya que la herramienta principal para ello ha sido usada la herramienta de Bootstrap.

En este archivo se crea una sombra combinada a la calculadora y se asigna una altura mínima al body.

```
1 #calculadora {  
2   box-shadow: 10px 2px 3px rgba(41, 73, 179, 0.05), 25px -16px #0b59cf;  
3 }  
4 body {  
5   min-height: 100vh;  
6 }
```

Captura 5 styles.css

### 3.3. app.js

Contiene la lógica operacional de la aplicación.

La función principal que contiene al resto es una función flecha que recoge como elemento el documento index.html y que ejecutará el resto de las funciones solamente una vez el documento haya sido cargado por completo.

```
4  
5 /*Función que se ejecutará al cargar por completo el documento HTML */  
6 $(document).ready(() => {
```

Captura 6 App.js - Función con listener del documento HTML

#### 3.3.1 Declaración de variables

La variable operación es de tipo String, cambia su valor en función de la operación aritmética a realizar

```
7 /*Declaración de las variables*/  
8 let operacion = "";  
9 let num1 = 1.0;  
10 let num2 = 1.0;  
11 let resultado = "";
```

Captura 7 App.js Declaración de variables.

### 3.3.2 enableSegundoCampo() y disableSegundoCampo()

Activan o desactivan el segundo campo y cambian la opacidad de este si estuviera desactivado.

```
const enableSegundoCampo = () => {  
  $("#segundo-num").removeClass("opacity-50");  
  $("#formGroupExampleInput2").removeAttr("disabled");  
};  
  
/* ...  
  
const disableSegundoCampo = () => {  
  $("#segundo-num").addClass("opacity-50");  
  $("#formGroupExampleInput2").prop("disabled", true);  
};
```

Captura 8 App.js - enableSegundoCampo() y disableSegundoCampo()

### 3.3.3 desactivarBtns() y activarBtns ()

Funciones que desactivan o activan todos los elementos afectados por la clase "btnOperacion" los botones operacionales.

```
/* ...  
  
const desactivarBtns = () => {  
  $(".btnOperacion").prop("disabled", true);  
};  
  
/* ...  
  
const activarBtns = () => {  
  $(".btnOperacion").removeAttr("disabled");  
};
```

Captura 9 App.js desactivarBtns() y activarBtns()

### 3.3.4 recogerNums ()

Función la cual retorna false si tras asignar a num1 y num2 los valores de los campos de texto alguno fuera un NaN. Si todo estuviera en orden retorna true.

```
const recogerNums = () => {  
  if (  
    operation === "+" ||  
    operation === "-" ||  
    operation === "x" ||  
    operation === ":"  
  ) {  
    num2 = parseFloat($("#formGroupExampleInput2").val());  
  }  
  num1 = parseFloat($("#formGroupExampleInput").val());  
  if (Number.isNaN(num1) || Number.isNaN(num2)) {  
    return false;  
  } else {  
    return true;  
  }  
};
```

Captura 10 App.js - recogerNums()

### 3.3.5 operar ()

Función la cual opera y crea un String que es asignado a “resultado” en función del valor asignado a la variable “operación”.

```
const operar = () => {  
  if (operacion === "+") {  
    resultado = `${num1} + ${num2} = ${num1 + num2}`;  
  } else if (operacion === "-") {  
    resultado = `${num1} - ${num2} = ${num1 - num2}`;  
  } else if (operacion === "x") {  
    resultado = `${num1} x ${num2} = ${num1 * num2}`;  
  } else if (operacion === ":") {  
    resultado = `${num1} : ${num2} = ${num1 / num2}`;  
  } else if (operacion === "√") {  
    resultado = `√${num1} : ${Math.sqrt(num1)}`;  
  } else if (operacion === "Seno") {  
    resultado = `Seno de ${num1} : ${Math.sin(num1)}`;  
  } else if (operacion === "Coseno") {  
    resultado = `Coseno de ${num1} : ${Math.cos(num1)}`;  
  } else if (operacion === "Tangente") {  
    resultado = `Tangente de ${num1} : ${Math.tan(num1)}`;  
  } else {  
    resultado = `Cotangente de ${num1}: ${1 / Math.tan(num1)}`;  
  }  
};
```

Captura 11 App.js - operar()

### 3.3.6 annadirResultado ()

Cambia el texto al párrafo que se encuentra debajo de la calculadora por el String de “resultado”.

```
const annadirResultado = () => {  
  $("#muchotexto").text(resultado);  
};
```

Captura 12 App.js - annadirResultado ()

### 3.3.7 appenAlHistorial ()

Añade el String de “resultado” al párrafo que se encuentra debajo del resultado.

```
const appenAlHistorial = () => {  
  $("#historial").prepend(`${resultado}<br>`);  
};
```

Captura 13 App.js - appendAlHistorial()



### 3.3.8 mostrarResultado()

Función la cual llama a las anteriores funciones nombradas si los datos introducidos son válidos:

- recogerNums()
- operar()
- annadirResultado()
- appenAlHistorial()

Seguidamente, resetea los valores introducidos en los campos de texto y llama a disableSegundoCampo().

Puede ser llamada desde el botón de igual o el botón de: raíz, seno, coseno, tangente y cotangente

Vuelve a bloquear el segundo campo de texto.

```
const mostrarResultado = () => {  
  let seguir = recogerNums();  
  if (seguir && operacion !== "") {  
    operar();  
    annadirResultado();  
    $("#formGroupExampleInput").val("");  
    $("#formGroupExampleInput2").val("");  
    appenAlHistorial();  
  } else {  
    annadirResultado("Datos no válidos");  
    $("#formGroupExampleInput").val("Dato no válido");  
    $("#formGroupExampleInput2").val("");  
  }  
  disableSegundoCampo();  
  operacion = "";  
};
```

Captura 14 App.js - mostrarResultado()

### 3.3.9 Click listeners

Escuchan si el botón del que recogen el id es pulsado en algún momento. Cambian el valor de "operacion" y actúan en consecuencia dependiendo de si la operación necesita de un segundo valor, activando el segundo campo y desactivando el resto de botones operacionales; o mostrando el resultado directamente en caso de no necesitar un segundo campo.

```
// ----- EVENTS LISTENERS -----  
  
/*Escucha cuando se pulsa "btnSumar" y cambia la operación al valor "+", activa el segundo campo */  
$("#btnSumar").click(() => {  
  operacion = "+";  
  enableSegundoCampo();  
  desactivarBtns();  
});
```

Captura 15 App.js - Click listeners 1

```

/*Escucha cuando se pulsa "btnRestar" y cambia la operación a "-", activa el segundo campo */
$("#btnRestar").click(() => {
  operacion = "-";
  enableSegundoCampo();
  desactivarBtns();
});

/*Escucha cuando se pulsa "btnMultiplicar" y cambia la operación a "x", activa el segundo campo */
$("#btnMultiplicar").click(() => {
  operacion = "x";
  enableSegundoCampo();
  desactivarBtns();
});

/*Escucha cuando se pulsa "btnDividir" y cambia la operación a ":", activa el segundo campo */
$("#btnDividir").click(() => {
  operacion = ":";
  enableSegundoCampo();
  desactivarBtns();
});

/*Escucha cuando se pulsa "btnRaiz" y cambia la operación a "√", llama a mostrarResultado() */
$("#btnRaiz").click(() => {
  operacion = "√";
  mostrarResultado();
});

/*Escucha cuando se pulsa "btnSeno" y cambia la operación a "Seno:", llama a mostrarResultado() */
$("#btnSeno").click(() => {
  operacion = "Seno";
  mostrarResultado();
});

/*Escucha cuando se pulsa "btnCoseno" y cambia la operación a "Coseno:", llama a mostrarResultado() */
$("#btnCoseno").click(() => {
  operacion = "Coseno";
  mostrarResultado();
});

/*Escucha cuando se pulsa "btnTan" y cambia la operación a "Tangente:", llama a mostrarResultado() */
$("#btnTan").click(() => {
  operacion = "Tangente";
  mostrarResultado();
});

/*Escucha cuando se pulsa "btnCotangente" y cambia la operación a "Cotangente:", llama a mostrarResultado() */
$("#btnCotangente").click(() => {
  operacion = "Cotangente";
  mostrarResultado();
});

/*Escucha cuando se pulsa "btnIgual", llama a mostrarResultado() */
$("#btnIgual").click(() => {
  mostrarResultado();
  activarBtns();
});

```

Captura 16 App.js - Click listeners 2

## Bibliografía (Enlaces consultados)

- Documentación aportada en clase
- Bootstrap 5 Cheatsheet: <https://bootstrap-cheatsheet.themeselection.com/>

## Índice de figuras

<b>CAPTURA 1 HTML-HEAD</b> .....	4
<b>CAPTURA 2 HTML BODY 1</b> .....	4
<b>CAPTURA 3 HTML BODY 2</b> .....	5
<b>CAPTURA 4 HTML BODY 3</b> .....	5
<b>CAPTURA 5 STYLES.CSS</b> .....	6
<b>CAPTURA 6 APP.JS - FUNCIÓN CON LISTENER DEL DOCUMENTO HTML</b> .....	6
<b>CAPTURA 7 APP.JS DECLARACIÓN DE VARIABLES.</b> .....	6
<b>CAPTURA 8 APP.JS - ENABLESEGUNDOCAMPO() Y DISABLESEGUNDOCAMPO()</b> .....	7
<b>CAPTURA 9 APP.JS DESACTIVARBTNS() Y ACTIVARBTNS()</b> .....	7
<b>CAPTURA 10 APP.JS - RECOGERNUMS()</b> .....	7
<b>CAPTURA 11 APP.JS - OPERAR()</b> .....	8
<b>CAPTURA 12 APP.JS - ANNADIRRESULTADO ()</b> .....	8
<b>CAPTURA 13 APP.JS - APPENDALHISTORIAL()</b> .....	8
<b>CAPTURA 14 APP.JS - MOSTRARRESULTADO()</b> .....	9
<b>CAPTURA 15 APP.JS - CLICK LISTENERS 1</b> .....	9
<b>CAPTURA 16 APP.JS - CLICK LISTENERS 2</b> .....	10

## Índice de tablas

<b>TABLA 1 FRONT END</b> .....	3
<b>TABLA 2 BACKEND</b> .....	3

## GLOSARIO

CFGS	Ciclo Formativo de Grado Superior
UD	Unidad Didáctica
HTML	Hypertext Markup Language
CSS	Cascading Style Sheet
ADAT	Acceso a datos