

Documentación Matriz.

El modelo empleado para esta App Web es MVC, aunque no es necesario el modelo ya que no trabajamos con ninguna base de datos y también se utilizó el framework materialize para el diseño web, el arbol de documentos es el siguiente:

root-

```
wwwroot-
  css-
    materialize.css
  fonts-
    roboto-
      fuentes
  js-
    materialize.js
    rotate.js
controller-
  HomeController.cs
Views-
  Home-
    Index.cshtml
  Shared
    _Layout.cshtml
    Error.cshtml
```

Metodos

Archivo: rotate.js

```
/*Esta función se llama cuando la página está lista y cargada en el browser*/
function ini()

/*El primer form que tiene los datos de la matriz no es necesario ir al servidor para
crear
el formulario de ingreso de cada elemento de la matriz, por eso se escucha el evento de
submit*/
$('body').on('submit', '#form-create', function(e)

//Antes de enviar los datos de la matriz al servidor, se transforman a JSON y se envían
por AJAX
$('body').on('submit', '#form-rotate', function(e)

//Evento para verificar que se introduzcan solamente numeros
$('body').on('keydown', 'input', function(e) {

/*Función que inicia el formulario para ingresar el numero de filas y columnas*/
function init_body()

/*Función para crear el layout del formulario para ingresar los datos de cada elemento
de la matriz*/
function create_input_matrix()

/*Función para imprimir las matrices*/
```

```
function print_matrices(matrix, matrix_input, matrix_output)
```

Archivo: HomeController.cs

```
/*Este metodo es el que manda a llamar la pagina de las matrices a traves de rotate.js  
con Ajax*/
```

```
public IActionResult rotate(string Omatrix)
```

```
/*Este metodo sirve para poder rotar la matriz*/  
private string[,] Rotate_Matrix(string[,] matrix)
```

```
/*Este metodo sirve para poder rotar la matrices que sean cuadradas*/  
private string[,] Rotate_Square_Matrix(string[,] matrix)
```