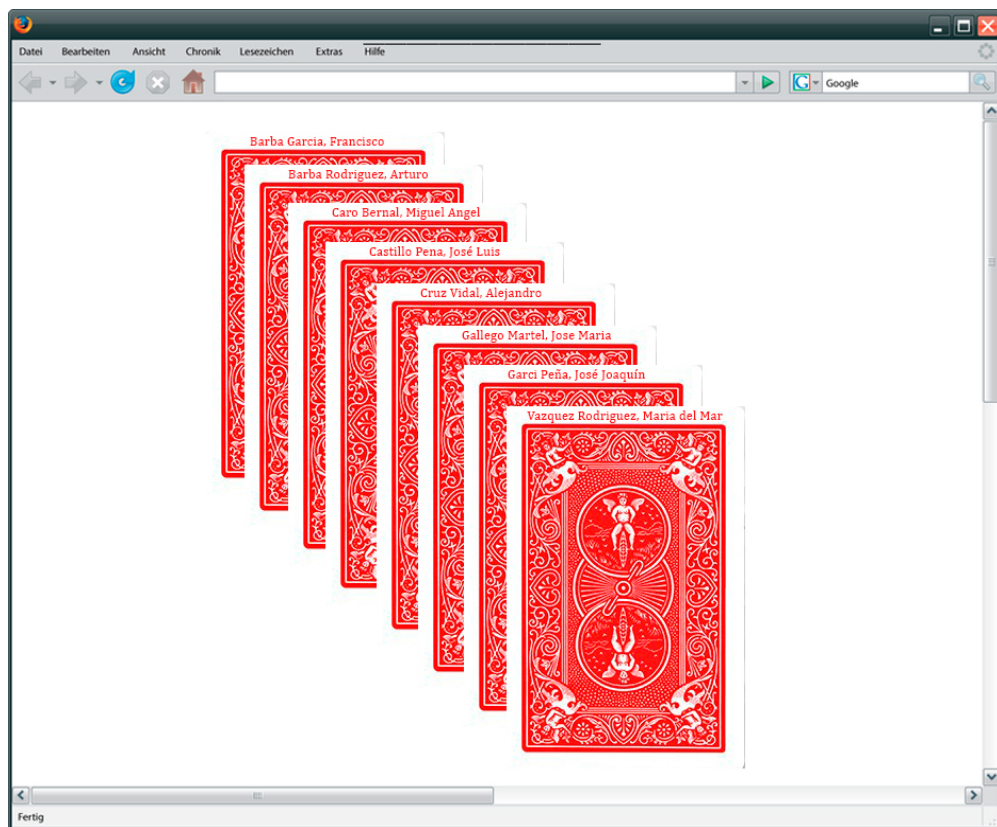


# Interface WEB

## Eventos y ubicación

Partiendo de un conjunto de cartas, hacer uso de eventos y lecturas de ubicación de elementos DOM para ordenarlas alfabéticamente. Todas las cartas deben quedar apiladas y desplazadas ligeramente ( $\pm 10\%$  de su tamaño). Una vez apiladas y ordenadas, la primera de ellas debe quedar en el fondo y cubierta por el resto (ver imagen).

Debe impedirse la superposición de la carta que no responda al orden, volviéndose al lugar que ocupaba anteriormente.



## Desarrollo de Aplicaciones Web - DISEÑO DE INTERFACES WEB

Actividad: Eventos y ubicación

Viernes, 25 Enero 2019

#### HTML:

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Superponiendo cartas</title>
    <link rel="stylesheet" href="css/master.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
    <script>window.jQuery || document.write('<script
src="js/jquery-3.3.1.min.js"></script>')</script>
    <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
    <script src="js/script.js" charset="utf-8"></script>
  </head>
  <body>
    <div id="des">

    </div>
    <div id="ord">

    </div>
  </body>
</html>
```

#### CSS:

```
#ord {
  z-index: -1;
  right: 0;
  top: 0;
  position: absolute;
  width: 50%;
  min-height: 750px;
  vertical-align: top;
  background-color: green;
}
img {
  width: 200px;
  left: 250px;
  position: absolute;
}
```

#### SCRIPT:

```
$(function(){
  loadImages();

  /**
   * Pone el div objetivo como droppable, siendo este el encargado de bloquear y encargarse
   * @param {Event} event Evento de drop.
   * @param {Event} ui Evento que causa el drop (el draggable).
   */
  $('#ord').droppable({drop: function(event, ui){
    var $carta = ui.draggable;
    var $this = $(event.target);
```

```

    var actual = $this.data('orden');
    var id = $carta.prop('id');
    if (!actual || actual < id) {
        $carta.draggable({revert: false});
        let $last = $('img[id=${actual}]')
        $carta.css(
            {
                'z-index': id,
                'left': `${parseInt($last.css('left'))+40}px`,
                'top': `${parseInt($last.css('top'))+40}px`
            }
        );
        $this.data('orden', id);
        $carta.draggable('destroy');
    }
    });
});

/**
 * Carga las imagenes en el navegador.
 */
function loadImages () {
    var img = [
        'src/Barba Garcia, Francisco.jpg',
        'src/Barba Rodriguez, Arturo.jpg',
        'src/Caro Bernal, Miguel Angel.jpg',
        'src/Castillo Peña, José Luis.jpg',
        'src/Cruz Vidal, Alejandro.jpg',
        'src/Gallego Martel, Jose Maria.jpg',
        'src/Garci Peña, José Joaquín.jpg',
        'src/Vazquez Rodriguez, Maria del Mar.jpg',
    ];

    setImgs(shuffle(initImgs(img)));
}

/**
 * Inicializa el array de imagenes con los valores basicos iniciales.
 * @param {Array} ar Array a inicializar de imagenes.
 * @return {Array} Array ya inicializado.
 */
function initImgs(ar) {
    var $elem = [];
    var $aux;
    for (var i = 0; i < ar.length; i++) {
        $aux = $('');
        $aux.draggable({revert: true});
        $elem.push($aux);
    }
    return $elem;
}

```

```

/**
 * Pone las imagenes en su div correspondiente.
 * @param {Array} ar Array de objetos a colocar.
 */
function setImgs(ar) {
    var $selector = $('#des');
    var pos = 10;
    for (var i = 0; i < ar.length; i++) {
        ar[i].css({'z-index': i, 'top': `${pos}px`, 'position': 'absolute'});
        pos+=60;
        $selector.append(ar[i]);
    }
}

```

```

/**
 * "Baraja" un array, haciendo que sea aleatorio
 * @param {Array} array Array a aleatorizar.
 * @return {Array} Array aleatorio.
 */
function shuffle(array) {
    var i = array.length, aux, random;

    while (0 !== i) {
        random = Math.floor(Math.random() * i);
        i -= 1;

        aux = array[i];
        array[i] = array[random];
        array[random] = aux;
    }

    return array;
}

```