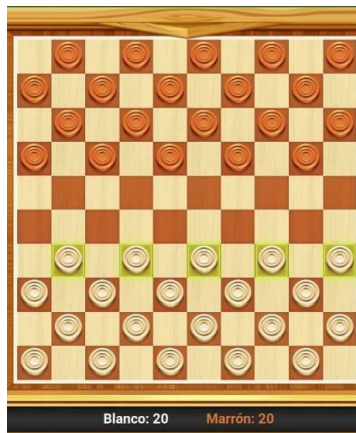


## Descripción de web app

Se desarrollará una aplicación que implemente el juego Damas, habrá una página donde los jugadores ingresarán su información y personalizarán la partida, después de esto iniciará el juego en una nueva página con el tablero de juego y la información respectiva para cada jugador.


El juego de Damas consiste en mover las piezas en diagonal a través de los cuadros negros de un tablero de 100 cuadros negros y blancos (10x10). Se juega entre dos personas, moviendo una ficha por turno, comiendo o capturando las fichas del rival pasando por encima de éstas, la intención del juego es dejar sin movimientos al rival o comer todas sus fichas.



Referencia: [damas](#)


## Diseño

	A	B	C	D	E	F	G	H	I	J	
10											10
9											9
8											8
7											7
6											6
5											5
4											4
3											3
2											2
1											1
	A	B	C	D	E	F	G	H	I	J	

**Jugador Norte**

Nombre:

Color:

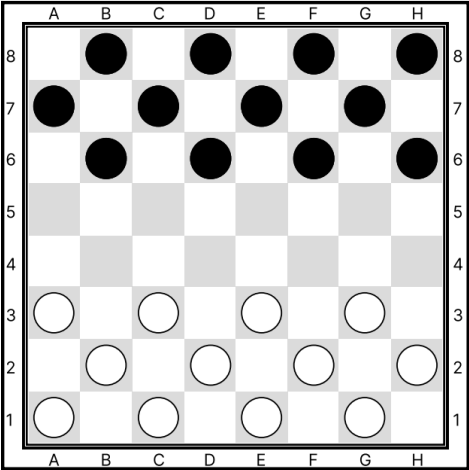
**Jugador Sur**

Nombre:

Color:

Tamaño tablero:

JUGAR

<b>Jugador Norte</b> Nombre: <input type="text"/> Nombre Color: <input type="text"/> Negro Movimientos: <input type="text"/> 12 Fichas: <input type="text"/> 0 Fichas capturadas: <input type="text"/> 0			<b>Jugador Sur</b> Nombre: <input type="text"/> Nombre Color: <input type="text"/> Blanco Movimientos: <input type="text"/> 12 Fichas: <input type="text"/> 0 Fichas capturadas: <input type="text"/> 0
<b>Información</b> Turno: <input type="text"/> Jugador Uno <input type="button" value="Rendirse"/>			

## Retrospectiva

Donde se usa mejor cada tecnología.

- **HTML**  
**index.html** → `10 <div id="inicio">`

En esta sección se hace uso de algunas etiquetas de HTML con el fin de recibir la información del juego de damas, entre las cuales tenemos **div** para agrupar a los dos jugadores, **img** donde se representa la ficha de cada jugador con el color correspondiente al que se tiene elegido, **label** que se usa para describir los elementos de entrada como el color de las fichas, **input** donde el usuario ingresa los nombres de los jugadores, **select** con el cual se despliega una lista con los colores y tamaños que están disponibles en el juego y un **button** con el cual se da inicio al juego.

Cada uno de las etiquetas anteriores tienen sus atributos respectivos entre los cuales están class, id, alt, src, name y style que son usados para los estilos de la página y las interacciones del usuario con esta.

- **CSS**  
**css/juego.css**

```
.jugador {
  margin: 10px;
  display: grid;
  padding: 5px;
  font-family: "Courier New", Courier, monospace;
  font-size: 18px;
  justify-self: center;
  align-items: center;
  justify-content: center;
}
```

```
    grid-template: repeat(6, 16.7%) / repeat(2, 50%);  
  }
```

Para mostrar la información de los jugadores se usa un selector de clase (.jugador) para este se le da un margen de 10 pixeles con **margin**, para determinar la forma en como se va a mostrar el elemento usamos **display** con un valor de grid para que sea un diseño de cuadrícula, junto con este tenemos **grid-template** para definir la estructura de las filas y columnas para lo cual se da un valor al cual corresponde a que se divide en 6 filas y 2 columnas. Además, se tiene un **padding** de 5 pixeles, la fuente de las letras y el tamaño se asignan con **font-family** y **font-size**. Por último, se centra horizontalmente los elementos de clase jugador con **justify-self** y se centran tanto vertical como horizontalmente los elementos dentro de este con **align-items** y **justify-content**.

- **JS**

js/vista.js → **28 VistaInicio.eventos()**

Se hace uso de **jQuery** para manipular el DOM, accediendo a los selectores para en el caso de que cambien se actualice la imagen de la ficha de cada jugador, para esto se accede a las imágenes y se cambia el valor para el atributo src. Además, se crea el evento para el botón jugar en el que al oprimirlo se obtienen los valores ingresados para los jugadores y se crea el juego de damas, de igual forma si al hacer esto hay algún error se mostrará en la pantalla con una alerta.

- **JS Testing**  
**Spec/JuegoSpec.js**

Con la librería **Jasmine** se hicieron las pruebas unitarias para probar las diferentes funcionalidades de la lógica de negocio, como la creación del tablero, mover las fichas, capturar fichas, cambio de peón a dama, etc.

- **JS Patterns**  
**js/**

Se realiza la separación de la lógica de presentación y la lógica de negocio (modelo.js, vista.js y controlador.js), lo que facilita la gestión y mantenimiento de la aplicación, además se hace uso de la librería knockoutjs con el mismo propósito mencionado anteriormente.