## **ÍNDICE MEMORIA**

[**ÍNDICE MEMORIA 1**](#_2g7shcx91caj)

[**1. Introducción 2**](#_zgqlcefv1k34)

[**2. Estructura general del documento 2**](#_g7ravxr122k4)

[**Esquema general 2**](#_yme4zkhhtoqf)

[**3. Sistema de diseño empleado 3**](#_ot658wm3loq)

[**3.1 Uso de Bootstrap 3**](#_nmylo89trqzh)

[**3.2 Uso de Flexbox 3**](#_s0h1wh44jywx)

[**4. Estilos CSS personalizados 3**](#_rycej7d1tsy6)

[**4.1 Colores base y casillas 3**](#_vl48k0prkofr)

[**4.2 Casillas especiales 4**](#_vh87fg5op51f)

[**5. El centro del tablero 4**](#_ds1wzt92rsi8)

[**5.1 Estructura interna 4**](#_muyvfm1p2udj)

[**5.2 Uso de degradados para los triángulos 4**](#_ot67262cfz0i)

[**5.3 Diagonales centrales negras 5**](#_3mozh8nw5fd5)

[**6. Dificultades encontradas 5**](#_byktdw73hpph)

[**6.1 Alineación del centro 5**](#_zg82imilex9k)

[**6.2 Transiciones entre zonas 5**](#_hc3qzpeokd95)

[**6.3 Escalabilidad 5**](#_x1byzjmf4sqz)

[**7. Conclusión 6**](#_f37aivsuv4c7)

[**8. Imágenes 6**](#_gyml48aclhzz)

[**Parchis 6**](#_wcfbd4b6pxq4)

[**Primer row 7**](#_h3oengqn00p7)

[**Segundo row 8**](#_9559zy4u20r2)

[**Tercer Row 9**](#_2cf4nribpumx)

[**Cuarto Row 10**](#_s7shh7ozzw2e)

[**Quinto row 11**](#_9qrv31p99l7v)

## 

## **1. Introducción**

El objetivo de este proyecto ha sido recrear el tablero clásico del juego de Parchís utilizando únicamente HTML y CSS, sin recurrir a imágenes ni gráficos externos.  
El propósito principal fue lograr una representación visual fiel y proporcionada del tablero empleando estructuras de cuadrícula y subdivisiones flexibles, haciendo uso del sistema grid de Bootstrap y propiedades de diseño flexibles (Flexbox).

Además, se ha buscado que el código sea ordenado, escalable y semánticamente claro, evitando soluciones improvisadas o “ñapas” que comprometan la estructura.

## **2. Estructura general del documento**

El archivo está formado por un documento HTML5 estándar con integración de Bootstrap 5.3.8.  
La estructura principal del tablero se organiza dentro de un contenedor (<div class="container">) dividido en cinco filas principales (1 superior, 3 centrales y 1 inferior), cada una compuesta por varias columnas que representan las zonas de las “casas” de cada color, los caminos y el centro del tablero.

### **Esquema general**

* Primera fila: zona superior (casa roja, casillas verticales y casa verde).
* Segunda, tercera y cuarta fila: zona central (con subdivisiones horizontales y el centro del tablero).
* Quinta fila: zona inferior (casa azul, casillas verticales y casa amarilla).

Cada celda del tablero se representa mediante <div> con clases que definen su color, borde y altura con sus respectivos estilos CSS.

## **3. Sistema de diseño empleado**

### **3.1 Uso de Bootstrap**

Bootstrap se ha empleado fundamentalmente para:

* Aprovechar su sistema de columnas (col-\*).
* Mantener la proporcionalidad del tablero sin necesidad de definir manualmente anchos en píxeles.
* Facilitar la división del tablero en 12 columnas, lo que permite repartir de forma homogénea las secciones (col-3 para cada casa de esquina y col-1 para columnas intermedias).

### **3.2 Uso de Flexbox**

Flexbox se ha utilizado de manera intensiva en:

* La alineación vertical y horizontal de los elementos (casillas grises de zona segura y en las casillas de salida).
* Las subdivisiones dentro de las filas y columnas centrales, donde se requería colocar varias casillas ubicadas verticalmente o distribuidas horizontalmente.

## **4. Estilos CSS personalizados**

### **4.1 Colores base y casillas**

Se definieron clases para los cuatro colores principales del juego:

.rojo { background-color: red; }

.verde { background-color: limegreen; }

.amarillo { background-color: gold; }

.azul { background-color: deepskyblue; }

Cada casilla cuenta con un borde negro (border: 2px solid black) y una altura uniforme (height: 40px) para mantener la coherencia visual.

### **4.2 Casillas especiales**

* Casillas grises: representan las zonas seguras.  
  Se diseñaron como contenedores flex con un pseudo-elemento ::after que genera el punto blanco central.
* Casillas de salida: utilizan el mismo principio que las grises, pero con el color correspondiente y un círculo blanco en el centro gracias al pseudo-elemento.
* Casillas blancas: se usan para los tramos normales de recorrido y se les asigna numeración de referencia mediante texto alineado (text-align: left o text-align: right).

## **5. El centro del tablero**

Esta ha sido la parte más compleja y laboriosa del proyecto.

El centro se diseñó sin imágenes ni SVG, únicamente con subdivisiones de <div> y degradados CSS, realizando cuadrículas internas para cuadrar este.

### **5.1 Estructura interna**

El centro se compone de una cuadrícula de 6 columnas internas, cada una dividida verticalmente mediante flex-column.  
Cada fila está formada por dos “cuadrantes” horizontales (div.horizontal), dando lugar a una matriz de 6x2 que permite formar las diagonales de los cuatro triángulos centrales. Siendo así a su vez en conjunto las 3 filas una matriz 6x6.

### **5.2 Uso de degradados para los triángulos**

Para representar las diagonales que dividen los colores del centro, se utilizaron degradados lineales con distintos ángulos:

.centroDiagonalDerechaAbajo { background: linear-gradient(222deg, limegreen 50%, gold 50%); }

.centroDiagonalIzquierdaArriba { background: linear-gradient(222deg, red 50%, deepskyblue 50%); }

.centroDiagonalDerechaArriba { background: linear-gradient(138deg, red 50%, limegreen 50%); }

.centroDiagonalIzquierdaAbajo { background: linear-gradient(138deg, deepskyblue 50%, gold 50%); }

Estos degradados reproducen los triángulos del centro del Parchís, con precisión en la división del color (50%-50%), simulando el efecto visual de la estrella central.

### **5.3 Diagonales centrales negras**

Para marcar las líneas divisorias negras, se emplearon pseudo-elementos ::after sobre las clases .dividido-diagonalIzquierda y .dividido-diagonalDerecha, con linear-gradient y position: absolute;.  
Esto permitió crear la cruz negra sin superponer imágenes ni usar bordes innecesarios.

## **6. Dificultades encontradas**

### **6.1 Alineación del centro**

La principal dificultad residió en lograr que todas las casillas centrales y diagonales encajaran perfectamente dentro del sistema de filas y columnas de Bootstrap.  
Fue necesario ajustar:

* Los márgenes y paddings (p-0, m-0).
* Los bordes de cada cuadrante.
* La altura uniforme de los elementos (height: 100px en algunos casos).

Esto permitió que las líneas negras de separación y las diagonales coincidesen exactamente sin dejar huecos ni solapamientos sin el realizamiento de ñapas.

### **6.2 Transiciones entre zonas**

Otra complicación fue mantener la coherencia visual entre las zonas coloreadas y las blancas, especialmente en las esquinas donde convergen tres colores diferentes.  
Se resolvió aplicando bordes manuales (border-top, border-bottom, etc.) en cada cuadrante para lograr continuidad visual.

### **6.3 Escalabilidad**

Al trabajar con Bootstrap, el tablero es ligeramente adaptable, pero al tener medidas fijas en altura (height: 40px), el escalado completo requiere ajustes posteriores en CSS para pantallas pequeñas, no es responsive al 100%.

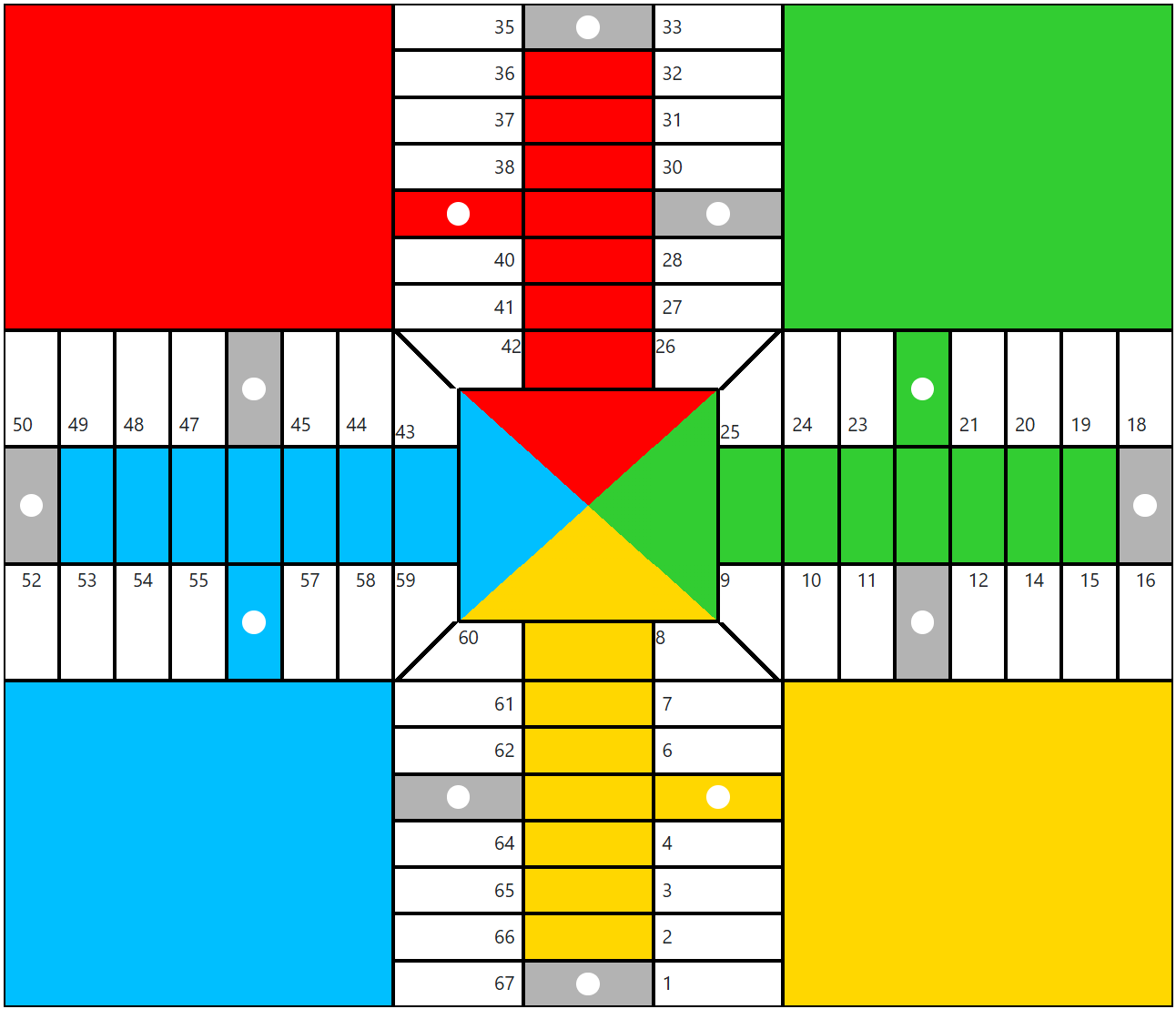
## **7. Conclusión**

El resultado final es un tablero de Parchís completamente funcional a nivel visual, construido exclusivamente con HTML y CSS, sin imágenes ni trucos externos.  
El proyecto demuestra un control del sistema de columnas de Bootstrap, Flexbox y el uso creativo de degradados lineales para generar figuras geométricas complejas.

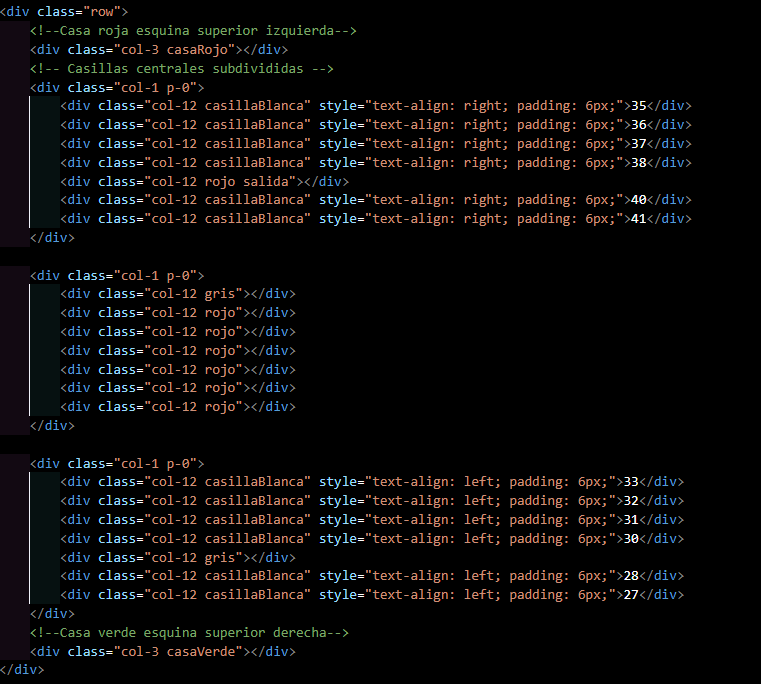
A pesar de la dificultad de la zona central, se ha logrado un diseño limpio, estructurado, cumpliendo el objetivo de realizar el tablero sin “ñapas” y con una disposición completamente generada por código.

## **8. Imágenes**

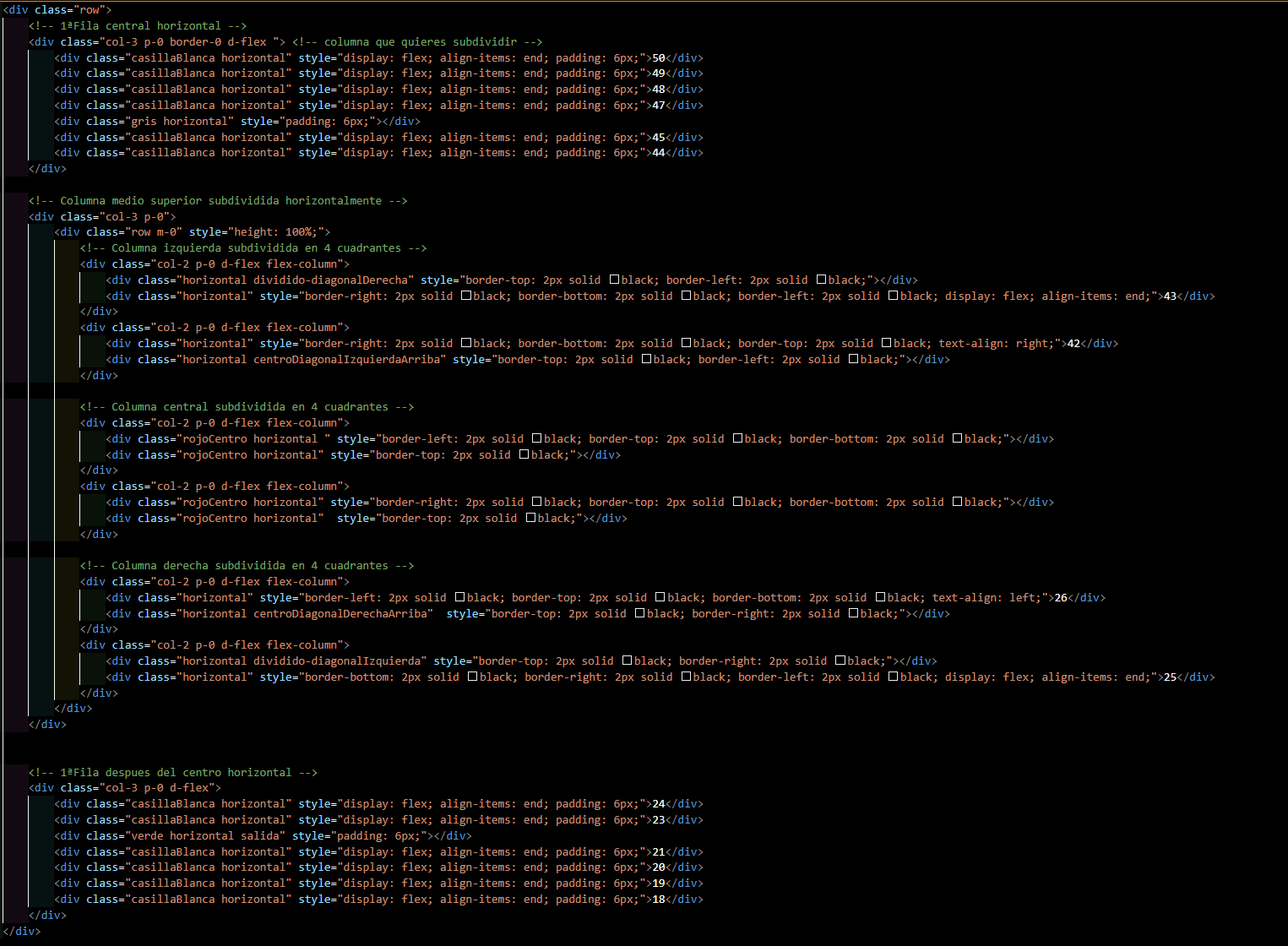
### **Parchis**



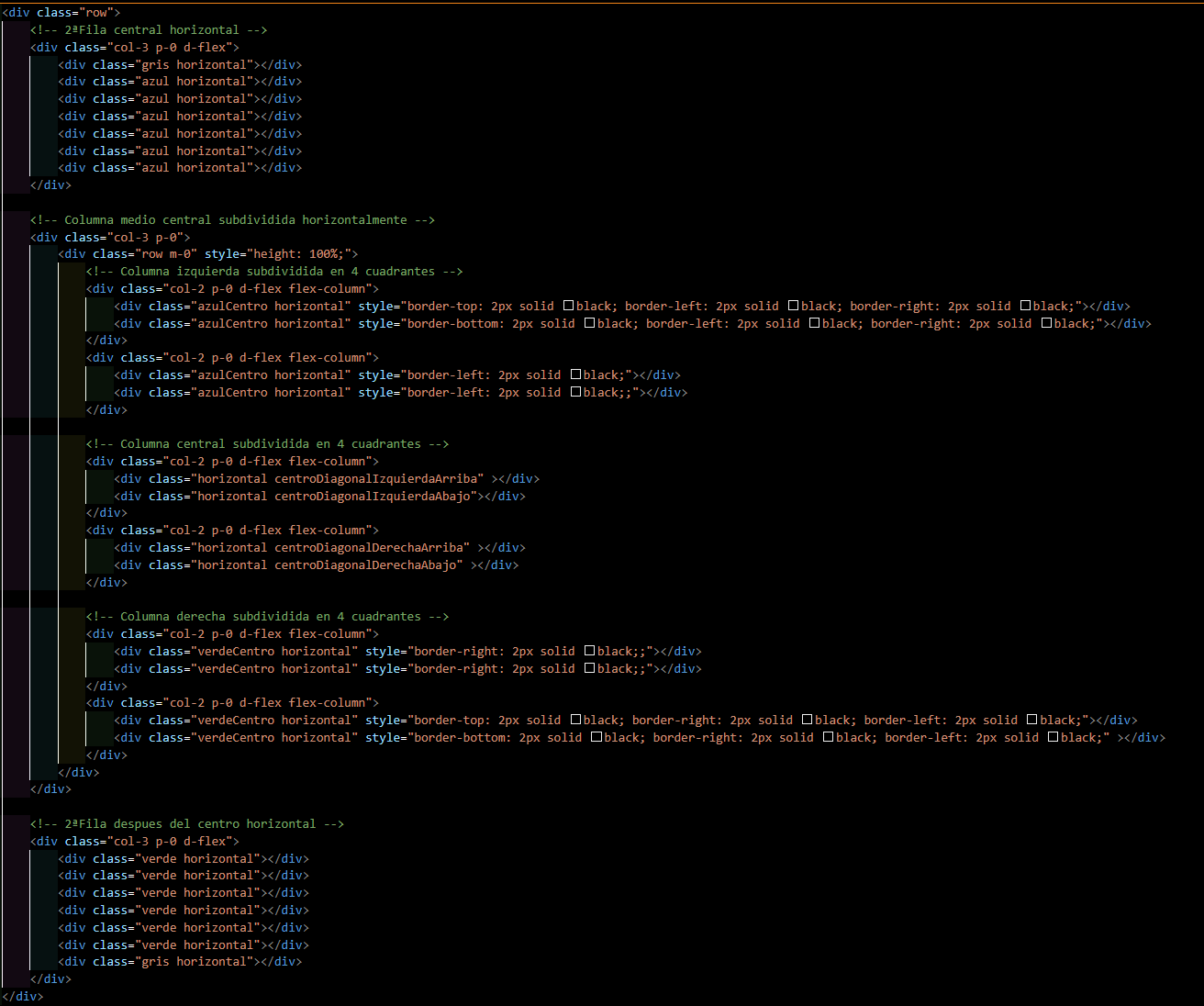
### **Primer row**



### **Segundo row**



### **Tercer Row**



### **Cuarto Row**



### **Quinto row**

