Entrega #1: UNO Web Edition

Reglas Básicas y UI Inicial

En esta entrega, su misión es unir estructura, presentación y lógica BÁSICA. Vamos a desarrollar la **versión básica del juego UNO**, 100% en el navegador, con JavaScript, HTML y CSS, con las **reglas básicas implementadas** y una **interfaz visual completa**, lista para jugar localmente entre jugadores simulados.

Requisitos obligatorios

Interfaz visual (HTML + CSS)

- Crear la **zona de descarte** y el **mazo de cartas**.
- Espacio para **los jugadores** y sus cartas visibles.
- Un **botón de "UNO"** para presionar cuando te queda una sola carta.
- Área de **notificaciones o mensajes** (por ejemplo, "¡Te saltaron!", "¡Robas 2!", etc.). Esto puede ser añadido para la entrega #2
- **Pantalla inicial de bienvenida** (estructura básica de una página dada en clases), con botón para iniciar partida. Agregar acá el nro de grupo y los nombres de los participantes. Sean creativos en esta sección. +2pts extra por creatividad.
- Aplicar **estilos visuales atractivos**:
    - Tipografía llamativa
    - Colores vivos
    - Disposición clara de zonas del juego
- Agregar **animaciones** (por ejemplo, al lanzar una carta).
- Incorporar **efectos de hover y transiciones** para mejorar la experiencia.

Lógica del juego (JavaScript)

- Repartir cartas (7 por jugador).
- Permitir lanzar una carta válida según color o número.
- Robar del mazo si no se puede jugar.
- Aplicar funcionalidades de:
    - Salto
    - Reversa
    - Roba +2
    - Comodín
    - Comodín +4
- Control de turnos (jugador actual y cambio de sentido).
- Validar el grito de “UNO” (y penalizar si no se pulsa el botón).
- Finalizar la ronda y mostrar qué jugador ganó.

Estructura de datos recomendada

Nota: La siguiente estructura NO dicta las únicas funciones, variables y constantes que deben ser añadidas. Sin embargo, es una guía perfecta para que manejen una estructura clara y reutilizable, ya que será clave cuando pasemos al desarrollo del entregable #2.

Variable globales:

```
let deck = [];           // Mazo de cartas por robar
let discardPile = [];    // Pila de cartas jugadas (descarte)
const colors = ['red', 'green', 'blue', 'yellow']; // Colores disponibles en el juego
const specialCards = ['jump', 'reverse', 'draw2']; // Cartas especiales
```

Variables de control del juego:

```
let players = [];           // Array con todos los jugadores (player, más abajo)
let currentPlayerIndex = 0; // Índice del jugador que tiene el turno actual
let direction = 1;          // Dirección del juego: 1 = horario, -1 = antihorario
```

Carta:

```
const card = {
  id: 'R-5',           // Código único: color + valor o tipo
  color: 'red',        // 'red', 'green', 'blue', 'yellow', 'wild'
  type: 'number',      // 'number' (numero), 'jump' (salto), 'reverse' (reversa), 'draw2' (roba2), 'wild' (comodin), 'wild+4' (comodin+4)
  value: 5             // Solo si type === 'number'
};
```

Jugador:

```
const player = {
    id: 'player1',        // Identificador único del jugador
    name: 'Stephanie',     // Nombre del jugador
    cards: [],           // Array de cartas en mano del jugador
    points: 0,           // Puntos acumulados del jugador
    saidUNO: false,      // Indica si el jugador dijo "UNO" cuando le queda 1 carta
    isHuman: true        // true si es jugador humano, false si es computadora
};
```

Estado inicial del juego:

```
const game = {
    players,              // Lista de jugadores
    deck,                 // Mazo de cartas
    discardPile,          // Pila de descarte
    turn: currentPlayerIndex, // Turno actual
    direction,            // Dirección del juego
    currentColor: null,   // Color actual del juego (se establece con comodines)
    waitingForColor: false, // Indica si se está esperando que se elija un color
    roundWinner: null     // Ganador de la ronda actual
};
```

Funciones recomendadas:

```
// Función para crear y mezclar el mazo inicial de UNO. Dinámico. Basado en los arreglos globales.
function initializeDeck() {...}

// Función para iniciar una nueva partida con el número especificado de jugadores
function startGame(numPlayers) { ... }

// Función para repartir las cartas iniciales a todos los jugadores
function dealCards() { ... }

// Función para que un jugador juegue una carta específica
function playCard(playerIndex, card) { ... }

// Función para que un jugador robe una carta del mazo
function drawCard(playerIndex) { ... }

// Función para pasar al siguiente turno (considerando la dirección del juego)
function nextTurn() { ... }

// Función para verificar si un jugador debe decir "UNO" (cuando le queda 1 carta)
function checkUNO(playerIndex) { ... }

// Función para contar los puntos de un jugador ganador
function countPoints(winnerIndex) { ... }

// Función para reiniciar una nueva ronda manteniendo los puntos acumulados
function resetRound() { ... }
```

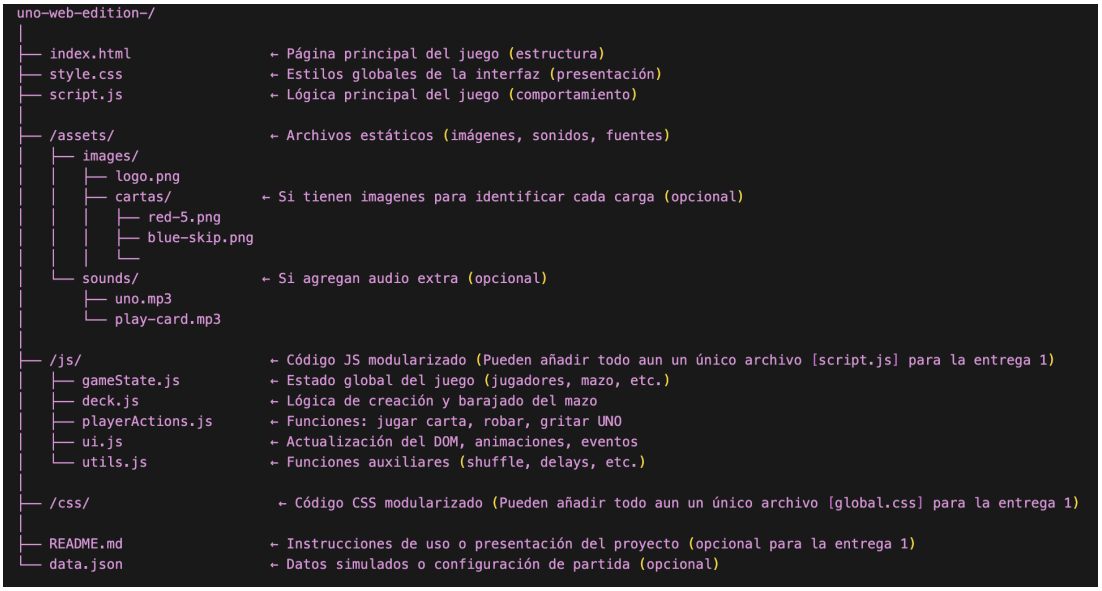
Criterios de evaluación

Criterio	Puntos
HTML + CSS	**10**
Estructura semántica: Uso correcto de etiquetas HTML5 (section, main, button, etc.).	0,5
CSS limpio y organizado: Uso de clases claras, separación lógica del layout	0,5
Zona del juego visible: Zona de descarte, mazo, cartas de jugadores, botón de UNO visibles.	4
Página de bienvenida: Incluye botón de iniciar, número de grupo y nombres. (+2pts extra por creatividad)	3,5
Estilo visual atractivo: Tipografía llamativa, colores vivos, disposición clara,hover en cartas, transiciones suaves, animaciones simples.	1,5
JavaScript	**10**
Reparto inicial: Reparto de 7 cartas por jugador y carta inicial en el descarte.	2
Jugada válida : Permitir lanzar carta por color o número.	2
Efectos de cartas : Función de cartas especiales (Salto, Reversa, Roba +2, Comodín, Comodín +4).	2
Robar y pasar turno: Si no hay carta válida, debe robar una y pasar turno.	1
Turnos funcionales : Flujo de turnos correcto, con cambio de sentido.	1
Botón de UNO : Validación del grito de "UNO" y penalización si no se presiona.	1
Fin de ronda: Detectar ganador de la ronda y reiniciar.	0,5
Código modular y claro : Uso de funciones reutilizables y estructura legible.	0,5

Nota: La nota final, dependerá de un interrogatorio realizado el día 30 de Junio.

Instrucciones

- Subir su proyecto en **`.zip`** o PREFERIBLEMENTE repo en GitHub con el nombre **GRUPO-NRO-X**.
- Árbol del proyecto ideal.



```
uno-web-edition-/
│
├── index.html                ← Página principal del juego (estructura)
├── style.css                 ← Estilos globales de la interfaz (presentación)
├── script.js                 ← Lógica principal del juego (comportamiento)
│
├── /assets/                  ← Archivos estáticos (imágenes, sonidos, fuentes)
│   ├── images/
│   │   ├── logo.png
│   │   ├── cartas/           ← Si tienen imagenes para identificar cada carga (opcional)
│   │   │   ├── red-5.png
│   │   │   ├── blue-skip.png
│   │   │   └──
│   └── sounds/               ← Si agregan audio extra (opcional)
│       ├── uno.mp3
│       └── play-card.mp3
│
├── /js/                      ← Código JS modularizado (Pueden añadir todo aun un único archivo [script.js] para la entrega 1)
│   ├── gameState.js          ← Estado global del juego (jugadores, mazo, etc.)
│   ├── deck.js               ← Lógica de creación y barajado del mazo
│   ├── playerActions.js      ← Funciones: jugar carta, robar, gritar UNO
│   ├── ui.js                 ← Actualización del DOM, animaciones, eventos
│   └── utils.js              ← Funciones auxiliares (shuffle, delays, etc.)
│
├── /css/                      ← Código CSS modularizado (Pueden añadir todo aun un único archivo [global.css] para la entrega 1)
│
├── README.md                 ← Instrucciones de uso o presentación del proyecto (opcional para la entrega 1)
└── data.json                 ← Datos simulados o configuración de partida (opcional)
```

  (Si el código está bien estructurado, la próxima fase (con WebSockets y una API) será mucho más fácil.)

- Fecha de entrega: **A más tardar el Domingo 29 de Junio.**
- Cualquier indicio del uso de la IA en el proyecto, será sancionado.

Recuerden que deben ser creativos. Agregar detalles visuales, sonidos, nombres, avatares, o cualquier otra solución, suma puntos.

Mucho éxito 🤗