

# MANUAL

José Luis Cotán Franco

José luis cotán franco  
2º DAW IES Doñana

## **Índice**

1ra parte – Elección de los Pokémon.....	2
2da parte – Elección del Pokémon inicial + Pantalla de juego .....	5
3ra parte – Modo y lógica del combate.....	7

## 1ra PARTE

Al iniciar el juego, se encontrará algo similar a:



El juego exige que cada jugador elija tres Pokémon diferentes (Se puede repetir entre jugadores), para ello se marcarán las casillas con los Pokémon deseados por cada jugador.

Para controlar que cada jugador no supere tres casillas marcadas, se usa el siguiente código:

```
// Limita la cantidad de casillas a 3.
function limitarSeleccion(fieldset) {
    // Selecciona todas las casillas de verificación.
    var checkboxes = fieldset.querySelectorAll('input[type="checkbox"]');

    // Itera sobre cada casilla de verificación.
    checkboxes.forEach(checkbox => {
        checkbox.addEventListener('change', function () {
            // Selecciona todas las casillas de verificación que están marcadas.
            var marcadas =
                fieldset.querySelectorAll('input[type="checkbox"]:checked');

            // Verifica si el número de casillas marcadas es superior a 3.
            if (marcadas.length > 3) {
                // Si es mayor que 3, desmarca la casilla actual.
                checkbox.checked = false;
            }
        });
    });
}

// Selecciona el 1er y 2do elemento fieldset del HTML
var fieldset1 = document.querySelector('fieldset:nth-of-type(1)');
var fieldset2 = document.querySelector('fieldset:nth-of-type(2)');

limitarSeleccion(fieldset1);
limitarSeleccion(fieldset2);
```

Una vez conseguido limitar hasta tres las casillas por cada jugador, necesitamos añadir los Pokémon seleccionados:

```
// Actualiza el array con los valores de los Pokémon seleccionados.
function actualizarPokemonsSeleccionados2() {
  // Se inicia como un array vacío.
  pokemonsSeleccionados2 = [];
  var checkboxes =
  // Selecciona todas las casillas de verificación con el nombre "pokemonJugador1" que estén marcadas.
    document.querySelectorAll('input[name="pokemonJugador2"]:checked');

  // Itera sobre cada casilla de verificación marcada.
  checkboxes.forEach(checkbox => {
    // Se obtiene el valor de la casilla marcada y se agrega ese pokemon al array.
    pokemonsSeleccionados2.push(Object.values(POKEMONS2)[checkbox.value]);
  });
}
```

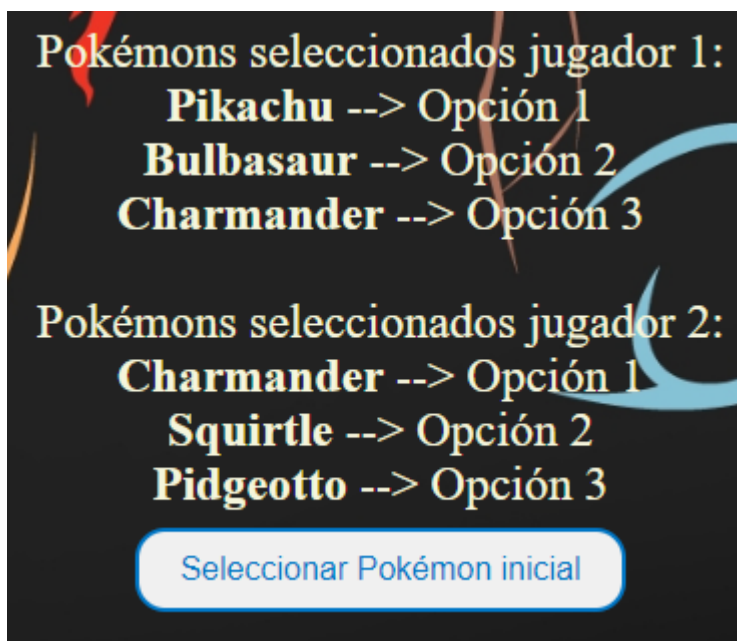
Datos a tener en cuenta:

- pokemonsSeleccionados2 es una variable global declarada anteriormente, pero se inicia como array vacío para que así no se añadan más Pokémon de los necesarios.
- POKEMONS2 es una constante del objeto pokemons, que contiene todos los Pokémon y sus datos

Con el siguiente código se actualiza los Pokémon seleccionados, en caso de que un jugador quiera rectificar

```
checkboxes.forEach(checkbox => {
  checkbox.addEventListener('change', actualizarPokemonsSeleccionados1);
});
```

Una vez marcado todos los Pokémon, le damos al botón que nos muestra los Pokémon, quedando un resultado similar a:



Esto se obtiene con el siguiente código:

```
// Agrega un evento al botón con ID "mostrarValores".
document.getElementById('mostrarValores').addEventListener('click', function () {
    //Se obtiene la referencia al elemento con el ID 'valoresMostrados'.
    var valoresMostrados = document.getElementById('valoresMostrados');

    // Se limpia el contenido existente.
    valoresMostrados.innerHTML = "";

    // Se muestra los Pokémon seleccionados por el jugador 1.
    valoresMostrados.innerHTML += "Pokémons seleccionados jugador 1:<br>";
    pokemonsSeleccionados1.forEach(function (pokemon, i) {
        valoresMostrados.innerHTML += `<strong>${pokemon.nombre}</strong>
        --> Opción ${i + 1}<br>`;
    });

    // Se agrega un espacio en blanco en 'valoresMostrados'.
    valoresMostrados.innerHTML += "<br>";

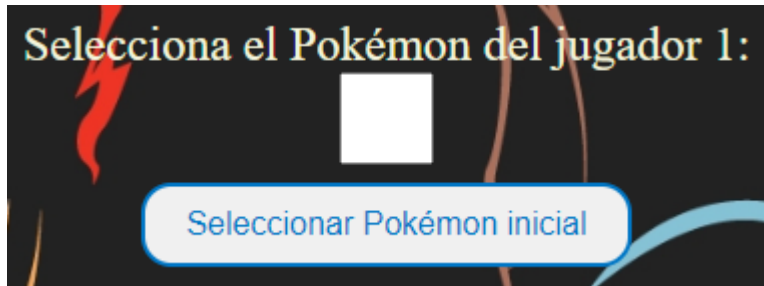
    valoresMostrados.innerHTML += "Pokémons seleccionados jugador 2:<br>";
    pokemonsSeleccionados2.forEach(function (pokemon, i) {
        valoresMostrados.innerHTML += `<strong>${pokemon.nombre}</strong>
        --> Opción ${i + 1}<br>`;
    });

    // Si ambos jugadores han seleccionado tres Pokémon cada uno,
    // muestra un botón para seleccionar un Pokémon inicial.
    if (pokemonsSeleccionados1.length === 3 &&
        pokemonsSeleccionados2.length === 3) {
        valoresMostrados.innerHTML += "<button id='seleccionarIniciales' \
        onclick='inicial()'>Seleccionar Pokémon inicial</button>";
    }
});
```

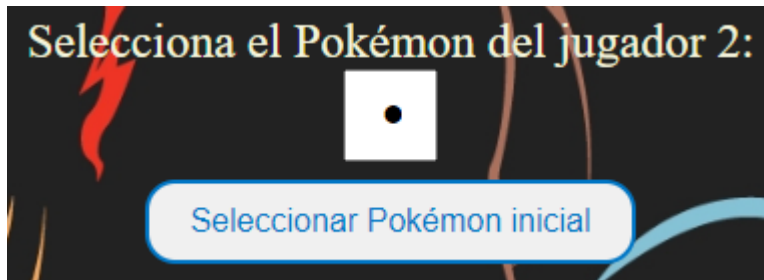
Con el último condicional "if", obligamos que ambos jugadores seleccionen tres Pokémon, en caso contrario no se mostrará el botón que avanza con el juego.

## 2ra PARTE

Se le pedirá a cada jugador que teclee su Pokémon inicial, mostrado anteriormente su número de referencia



Se oculta el valor introducido para evitar trampas



Se obtiene con el siguiente código (ejemplo de jugador 1):

```
function inicial() {  
    var valoresMostrados = document.getElementById('valoresMostrados');  
    if (jugadorActual === 1) {  
  
        // Se elimina el botón que muestra los Pokémon y pide a los jugadores sus Pokémon iniciales.  
        document.getElementById('mostrarValores').remove()  
        valoresMostrados.innerHTML = `Selecciona el Pokémon del jugador  
        ${jugadorActual}: <input type="password" id='seleccion'>`;   
        valoresMostrados.innerHTML += "<button id='seleccionarIniciales' \\  
        onclick='inicial()'>Seleccionar Pokémon inicial</button>";  
        document.getElementById('seleccion').style.display = 'block';  
        // Llama a la función, que obtendrá el Pokémon inicial elegido por el jugador  
        document.getElementById('seleccion').addEventListener('input', function() {  
            guardarPokemonInicial(1);  
        });  
        jugadorActual++;  
    }  
}
```

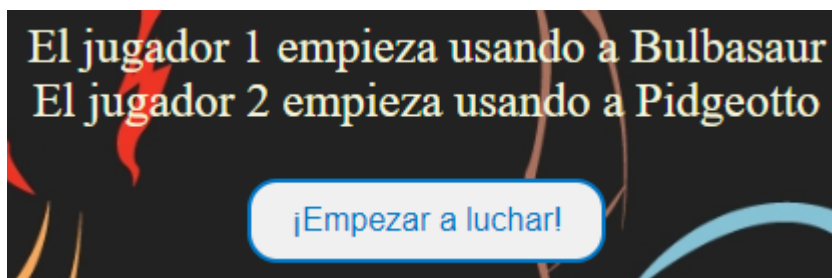
```
function guardarPokemonInicial(jugador) {
  var seleccionInput = document.getElementById('seleccion');
  var seleccion = document.getElementById('seleccion').value;

  if (isNaN(seleccion) || seleccion < 1 || seleccion > 3) {
    seleccionInput.value = '';
    return;
  }

  if (jugador === 1) {
    pokemonInicial1 = seleccion - 1;
    pokemonActual1 = seleccion - 1;
  } else if (jugador === 2) {
    pokemonInicial2 = seleccion - 1;
    pokemonActual2 = seleccion - 1;
  }
}
```

Se obliga al jugador a teclear 1, 2 o 3. En caso contrario, el input se restablece a "", impidiendo al jugador introducir un dato inválido. En caso de que el jugador avance sin teclear nada, el juego asignará como Pokémon inicial el 1ro de la lista.

Cuando ambos jugadores hayan elegido su Pokémon inicial, saldrá en pantalla algo similar a:



Esto se obtiene con el siguiente código:

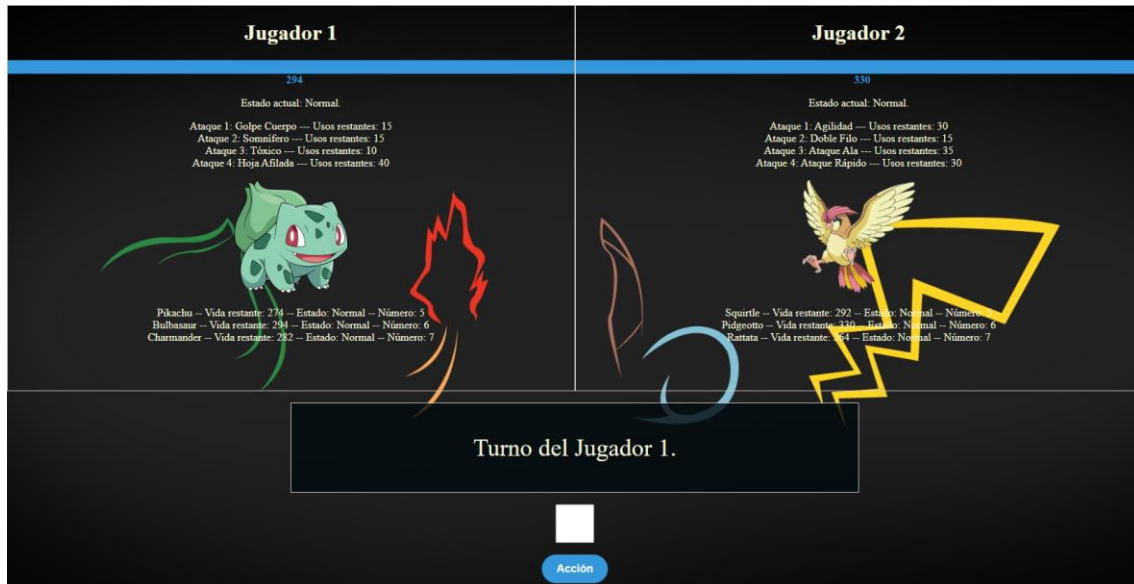
```
} else {
  // Muestra los Pokémon iniciales seleccionados para cada jugador.
  valoresMostrados.innerHTML = `El jugador 1 empieza usando a
  | ${pokemonsSeleccionados1[pokemonInicial1].nombre}`;
  valoresMostrados.innerHTML += `<br>El jugador 2 empieza usando a
  | ${pokemonsSeleccionados2[pokemonInicial2].nombre}`;
  // Muestra el botón de empezar el juego.
  botonEmpezar.style.display = 'inline-block';
  // Crea una copia profunda (sin referencia) de los Pokémon seleccionados
  pokemonsEstadisticasJugador1 = structuredClone(pokemonsSeleccionados1);
  pokemonsEstadisticasJugador2 = structuredClone(pokemonsSeleccionados2);
}
```

Se encuentra dentro de la función inicial().

Es necesario usar structuredClone() para crear una copia profunda del objeto, evitando conflicto en caso de que ambos jugadores eligiesen el mismo Pokémon.

## 3ra PARTE

Al darle click al botón que dice “¡Empezar a jugar!”, cargará una pantalla similar a:



Para obtener esto, es necesario cambiar el estilo de presentación de los elementos HTML.

```
document.getElementById('empezar').addEventListener('click', function () {  
    // Muestra el menú completo de combate  
    combates.style.display = 'flex';  
    menuPantalla.style.display = 'flex';  
    menuInput.style.display = 'flex';  
    menuBoton.style.display = 'flex';  
    pantalla.innerHTML = 'Turno del Jugador 1.';  
    document.getElementById('inicial').innerHTML = '';  
  
    // Muestra la información de los Pokémon de ambos jugadores  
    mostrarVida();  
    mostrarEstado();  
    mostrarAtaques();  
    mostrarPokemonsRestantes();  
    mostrarImagenes();  
});
```

Añadimos todo lo anterior con `x.style.display = 'flex';`

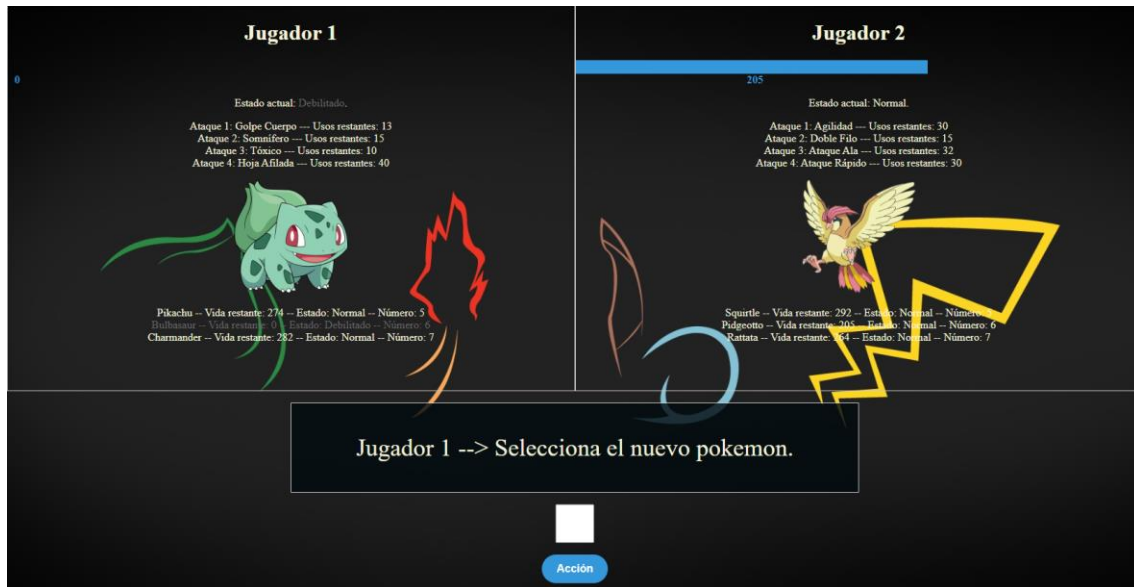
Para ocultar la lista de Pokémon, que estaba añadida dentro del div 'inicial', se establece su contenido HTML en "", usando `innerHTML`.



Para la lógica del combate, se han creado contadores.

Cuando el **contador es -1**, significa que un Pokémon ha sido debilitado y se avisa al jugador que debe cambiar su Pokémon.

Ejemplo dentro del juego:



Ejemplo del código:

Aquí se comprueba la vida de los dos Pokémon. Si el Pokémon es del jugador 1, el contador pasa a ser -1, además de darte los parámetros necesarios a otras variables.

```
if (pokemonUsuario1.estadisticas.vida <= 0 || pokemonUsuario2.estadisticas.vida <= 0) {  
  if (idJugador === 1) {  
    pantalla.innerHTML += `El Pokémon ${pokemonUsuario2.nombre} se ha debilitado<br>`;   
    pokemonUsuario2.estado = "Debilitado";  
    contAccion = -1;  
    pokemonDebilitado2 = true;  
    jugador1HaAtacado = true;  
    jugador2HaAtacado = true;  
  } else {
```

Aquí se le pide al usuario que cambie de Pokémon:

```
document.getElementById('hacerAccion').addEventListener('click', function () {  
  // Si un Pokémon ha sido debilitado, pide al jugador que seleccione el nuevo Pokémon  
  if (contAccion === -1) {  
    if (pokemonDebilitado1) {  
      pantalla.innerHTML = `Jugador 1 --> Selecciona el nuevo pokemon.`;  
      contAccion = 0;  
      return;  
    } else if (pokemonDebilitado2) {  
      pantalla.innerHTML = `Jugador 2 --> Selecciona el nuevo pokemon.`;  
      contAccion = 0;  
      return;  
    }  
  }  
})
```

Cuando el **contador es 0**, es para controlar las validaciones del jugador y actualizar los datos de los Pokémon.

Se valida si:

- El texto introducido por el usuario es un número
- El número tecleado es 5, 6 o 7.
- El Pokémon actual se encuentra debilitado.
- El Pokémon por el que se va a cambiar no se encuentra debilitado

Una vez se haya validado todo, se procede a cambiar el Pokémon y actualizar todos los datos en el juego.

```
} else if (contAccion === 0) {  
  let condicion1 = false;  
  let condicion2 = false;  
  let nuevoPokemonIndex = parseInt(document.getElementById('accion').value);  
  if ((nuevoPokemonIndex === 5 || nuevoPokemonIndex === 6 || nuevoPokemonIndex === 7)  
      && pokemonsSeleccionados1[nuevoPokemonIndex - 5].estado !== "Debilitado"  
      && pokemonDebilitado1) {  
    pokemonActual1 = nuevoPokemonIndex - 5;  
    eliminarImagenes();  
    mostrarImagenes();  
    mostrarEstado();  
    mostrarAtaques();  
    mostrarVida();  
    mostrarPokemonsRestantes();  
    pokemonDebilitado1 = false;  
  } else {
```

Cuando el **contador es 1**, es turno del jugador 1. Al jugador se le obliga a introducir un número entre 1 y 7 ambos inclusive:

```
else if (contAccion === 1) {  
  valorJugador1 = document.getElementById('accion').value - 1;  
  if (isNaN(valorJugador1) ||  
      parseInt(valorJugador1) < 0 ||  
      parseInt(valorJugador1) > 6) {  
    accion.value = '';  
    pantalla.innerHTML = "Introduce un número correcto.<br>\nTurno del Jugador 1.";  
    return;  
  }  
}
```

En caso de no introducir lo pedido anteriormente, se mostrará por pantalla el error, avisando al jugador que introduzca un número correcto.

Si el número introducido corresponde al mismo Pokémon que se está usando, no se permitirá avanzar.

```
if (parseInt(valorJugador1) > 3 && parseInt(valorJugador1) < 7) {  
    if (valorJugador1 - 4 === pokemonActual1) {  
        pantalla.innerHTML = "¡No puedes cambiar al mismo Pokémon!<br> \n  
        Turno del Jugador 1.";   
        accion.value = '';   
        return;   
    }  
}
```

Si el número introducido corresponde a un Pokémon que se encuentra debilitado, no se permitirá avanzar.

```
if (pokemonsSeleccionados1[valorJugador1 - 4].estado === "Debilitado") {  
    pantalla.innerHTML = "¡El Pokémon está debilitado!<br> \n  
    Turno del Jugador 1.";   
    accion.value = '';   
    return;   
}
```

Una vez introducido un valor correcto, se guarda en la variable "valorJugador1" y se suma 1 al conAccion.

Cuando el **contador es 2**, se usa la misma lógica que si el contador fuera 1.

Si el contador no es un número de los mencionados anteriormente, se procede a hacer los cambios o ataques realizado por los jugadores.

Primero se tiene preferencia al cambio de Pokémon, así que se hace primero. Aquí no se controla la velocidad, debido a que es indiferente.

Si el jugador eligió cambiar:

[illegible]

Se cambia de Pokémon, avisando y actualizando los datos en pantalla.

jugador1HaAtacado pasa a ser true, para controlar que el jugador 1 ha atacado.

Para el jugador 1, se usa la misma lógica.

Si el jugador decide atacar:

```

else if (compararVelocidad(pokemonsSeleccionados1[pokemonActual1],
pokemonsSeleccionados2[pokemonActual2]) &&
!jugador1HaAtacado) {
pantalla.innerHTML =
`El Pokémon ${pokemonsSeleccionados1[pokemonActual1].nombre}
del Jugador 1 ha usado
${pokemonsSeleccionados1[pokemonActual1].ataques[valorJugador1].nombre}.<br>`;
atacar(pokemonsSeleccionados1[pokemonActual1],
pokemonsSeleccionados2[pokemonActual2],
valorJugador1,
1);
jugador1HaAtacado = true;
return;
}

```

Primero se compara la velocidad de ambos Pokémon con la siguiente función:

```
function compararVelocidad(pokemon1, pokemon2) {  
  if (pokemon1.estadisticas.velocidad === pokemon2.estadisticas.velocidad) {  
    if (Math.floor(Math.random() * 101) > 50) {  
      return pokemon1.estadisticas.velocidad <  
        pokemon2.estadisticas.velocidad;  
    }  
  }  
  return pokemon1.estadisticas.velocidad > pokemon2.estadisticas.velocidad;  
}
```

Luego se usa la función atacar, con cuatro argumentos:

```
atacar(pokemonsSeleccionados1[pokemonActual1],  
pokemonsSeleccionados2[pokemonActual2],  
valorJugador1,  
1);
```

- **PokemonsSeleccionados1[pokemonActual1]**: pokemonsSeleccionados1 es un array que contiene los Pokémon seleccionados por el jugador 1, obtenido en la función actualizarPokemonsSeleccionados1()  
pokemonActual1 es el índice del Pokémon que se está usando actualmente, obtenido en la función guardarPokemonInicial(jugador) si es el 1er Pokémon o cuando el jugador cambia de Pokémon, en el contAccion = 0.
- **PokemonsSeleccionados2[pokemonActual2]**: La misma lógica de antes, pero para el jugador 2.
- valorJugador1: es el valor introducido anteriormente. Representa el índice de los ataques del Pokémon.
- 1: Representa el jugador, en este caso el jugador 1.

Fórmula de daño:

```
function obtenerDaño(pokemonUsuario1, pokemonUsuario2, ataque) {
  console.log(pokemonUsuario1.tipo.includes(ataque.tipo))
  res = 0.01 *
    (pokemonUsuario1.tipo.includes(ataque.tipo) ? 1.5 : 1) *
    (ataque.efectividad.some(tipo => pokemonUsuario2.tipo.includes(tipo)) ? 2 :
      (ataque.debilidad.some(tipo => pokemonUsuario2.tipo.includes(tipo))) ? 0.5 : 1) *
    (Math.floor(Math.random() * (16) + 85)) *
    (
      (21 * (ataque.tipoAtaque === "Físico" ?
        pokemonUsuario1.estadisticas.ataque * ataque.poder :
        pokemonUsuario1.estadisticas.ataqueEspecial * ataque.poder)) /
      (25 * (ataque.tipoAtaque === "Físico" ?
        pokemonUsuario2.estadisticas.defensa :
        pokemonUsuario2.estadisticas.defensaEspecial)) + 2
    )
  return res;
}
```

Representa la siguiente fórmula:

$$Daño = 0.01 \times B \times E \times V \times \left( \frac{(0.2 \times N + 1) \times A \times P}{25 \times D} + 2 \right)$$

- B = Bonificación. Si el Pokémon es del mismo tipo que el ataque que usa, se usará 1,5; en caso contrario, 1.
- E = Efectividad. Puede tomar los valores de 0, 0.5, 1, y 2.
- V = Variación. Es una variable que comprende todos los valores discretos entre 85 y 100 (ambos incluidos).
- N = Nivel del Pokémon que ataca, en este caso son todos de nivel 100.
- A = Cantidad de ataque o ataque especial del Pokémon. Si el ataque que utiliza el Pokémon es físico se toma la cantidad de ataque y si es especial se toma la cantidad de ataque especial.
- P = Poder del ataque, el potencial del ataque.
- D = Cantidad de defensa del Pokémon rival. Si el ataque que hemos utilizado es físico cogeremos la cantidad de defensa del Pokémon rival, si el ataque que hemos utilizado es especial, se coge la cantidad de defensa especial del Pokémon rival.

Si todos los Pokémon de un jugador son menor o igual a 0, habrá perdido, dando la victoria al otro jugador.

```
if (pokemonsSeleccionados1[0].estadisticas.vida <= 0 &&  
    pokemonsSeleccionados1[1].estadisticas.vida <= 0 &&  
    pokemonsSeleccionados1[2].estadisticas.vida <= 0) {  
    pantalla.innerHTML = "¡¡¡HA GANADO EL JUGADOR 2!!!";  
    menuInput.style.display = 'none';  
    menuBoton.style.display = 'none';  
    reiniciar.style.display = 'flex';  
    return;  
} else if (pokemonsSeleccionados2[0].estadisticas.vida <= 0 &&  
    pokemonsSeleccionados2[1].estadisticas.vida <= 0 &&  
    pokemonsSeleccionados2[2].estadisticas.vida <= 0) {  
    pantalla.innerHTML = "¡¡¡HA GANADO EL JUGADOR 1!!!";  
    menuInput.style.display = 'none';  
    menuBoton.style.display = 'none';  
    reiniciar.style.display = 'flex';  
    return;  
}
```

Aquí se comprueba. En caso de entrar en el condicional, desaparecerá el input y el botón de Acción, siendo este último reemplazado por un botón de reiniciar, que reiniciará el juego para una nueva partida.