



Contenido

Introducción:.....	2
Pantalla de inicio:.....	2
Inicio de combate:	2
Desarrollo del combate:	3
Secuencia de combate:.....	4
Resolución del enfrentamiento:.....	4
Animaciones estáticas de los personajes:.....	4
Animaciones de ataque de los personajes:.....	5
Prototipos que deberán aparecer:	5
Pokeball:.....	6
Criterios de calificación base:	7
Anexo I: Datos para el trabajo	8

Introducción:

El trabajo consiste en la simulación de un combate de la franquicia de juegos Pokémon usando JavaScript, el usuario controlará un entrenador con dos Pokemones (Pikachu y Ditto) contra un Pokémon gestionado por el ordenador (Articuno). El funcionamiento general de la página seguirá los siguientes puntos:

4 Marzo Entrega

Pantalla de inicio:

Se deberá programar una página inicial en la que, al hacer click, se iniciará el combate. El diseño de la misma se deja en manos del estudiante.

Inicio de combate:

El título desaparecerá y tras 3 segundos aparecerán (en este orden):

- Las plataformas del combate.
- Articuno en la plataforma opuesta.
- Pikachu (en el lado controlado por el usuario) como se muestra en la siguiente imagen:



-Dos divs que marcaran el nombre, puntos de salud y nivel de cada Pokémon.

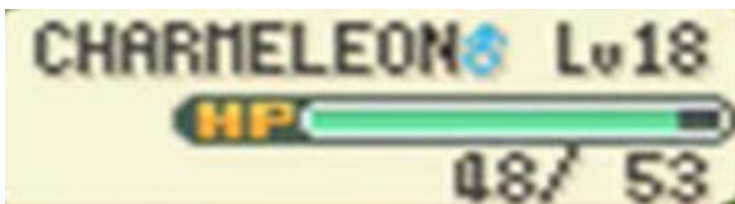


Imagen de referencia.

-Un menú de texto y un cuadro de opciones inferior que estará compuesto por dos divs.

What should PIKACHU do?	FIGHT	BAG
	POKEMON	RUN

Desarrollo del combate:

En cada turno el jugador podrá elegir entre:

- Luchar (Fight)
- Abrir la mochila (Bag) si quedan objetos.
- Cambiar de pokemón (Pokemon) si aún queda algún otro pokemón vivo.
- Tratar de huir (Run).

Si elige la bolsa, en caso de que queden objetos, mostrará en el div la opción de seleccionar uno (haciendo click). En caso contrario aparecerá un mensaje indicando: “The bag is empty!” y el usuario deberá elegir otra acción.

Si elige Pokemon, si quedá algún otro pokemón distinto al que está luchando, aparecerán 6 divs dividiendo el cuadro de texto en dos columnas de 3, con cada casilla los datos del pokemón en el siguiente formato:

Nombre del pokemón	Nivel
VidaActual/VidaMáxima	

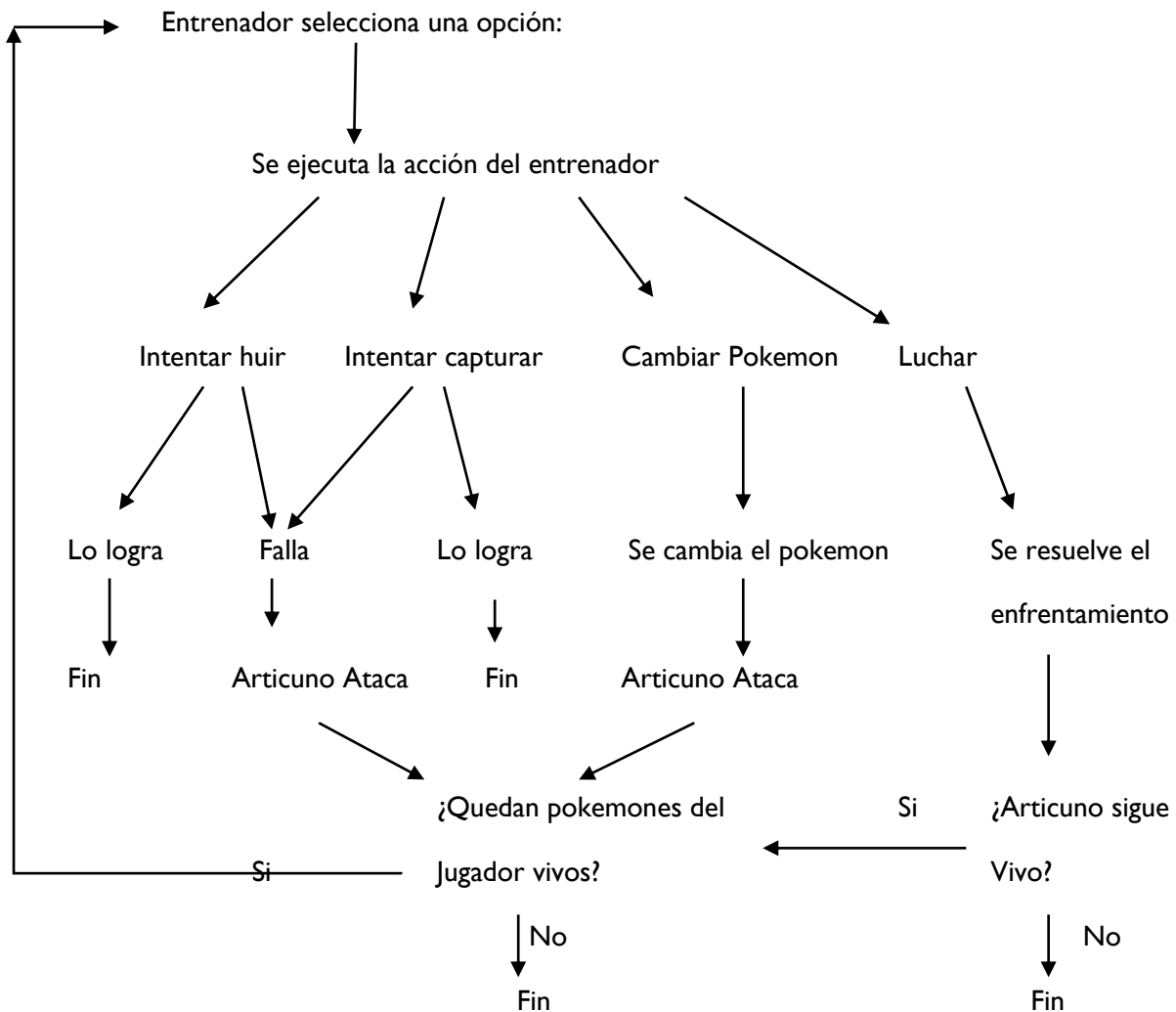
Teniendo un color de fondo rojo si la vidaActual es 0, y azul en otro caso. Sino quedan pokemones aparecerá el siguiente mensaje de texto en vez de los divs: “You are with your only Pokemon left!” y el usuario deberá elegir otra acción.

Si elige huir, intentará huir con un 20% de probabilidades de escape. En caso de lograrlo saldrá la pantalla final (similar que la de título pero con el mensaje: “You run...”). En caso de fallar, el pokemón del jugador pierde el turno y Articuno atacará.

Si elige luchar, aparecerá en el cuadro de texto un menú de opciones compuesto por los movimientos disponibles (4 divs también, independientemente de si tiene 4 movimientos) ejecutándose en la secuencia de combate aquel que seleccione (solo se pueden seleccionar aquellos que tienen PP no nulos y sobrevive).

Secuencia de combate:

4



Resolución del enfrentamiento:

En un enfrentamiento ambos pokemones están tratando de usar un movimiento, el primero en actuar dependerá de su atributo **Velocidad** (mejora 2, sino ejecuta siempre primero la opción del jugador y después una aleatoria de articuno). El segundo actuará solo si está vivo el pokemón.

En caso de que un pokemón se quede a cero puntos de vida el jugador intentará sacar a su siguiente pokemón. Sino queda ninguno, perderá. En otro caso, lo saca antes de elegir la acción a ejecutar.

Animaciones estáticas de los personajes:

-Mientras el usuario esté eligiendo que hacer, ambos pokemones se desplazarán verticalmente una pequeña distancia antes de volver a bajar.

Animaciones de ataque de los personajes:

-Desde justo antes de que se comience a ejecutar la acción del entrenador y hasta que sea el turno de elegir una opción del entrenador ambos personajes estarán quietos en escena por defecto.

-Cuando uno de los pokemones ejecute una habilidad deberá desplazarse rápidamente una pequeña distancia hacia el otro pokemón para volver al momento a su posición. Hasta que el primer pokemón no vuelva, el siguiente no debe iniciar su animación de ataque (si sigue vivo).

Prototipos que deberán aparecer:

Pokemon:

Atributos (al menos): HP, Lvl, Ataque, Defensa, movimientos (que será un array compuesto por objetos de tipo movimiento), tipo 1 y a veces tipo 2.

1ª Mejora: Añadir ataque especial y defensa especial y modificar el cálculo del daño de manera acorde.

2ª Mejora: Añadir Velocidad y modificar de manera acorde el orden de los turnos.

3ª Mejora: Estado y Habilidades de acuerdo a lo indicado más abajo.

Deberá incluir también aquellos métodos que se consideren necesarios.

Movimiento:

Atributos: Potencia, Precisión, PP, Objetivo, Tipo

1ª Mejora: Categoría (podrá ser una entre Físico, Especial y Estatus, no hay otra posibilidad y deberá verse reflejado en el código)

3ª Mejora: Efectos: Congelado (20% Probabilidad de descongelarse en cada turno, sino durante el enfrentamiento pierde el turno) y Parálisis (Reduce la velocidad un 25% y tiene un 25% de probabilidad de perder el turno).

Deberá incluir los métodos necesarios.

La probabilidad de que un ataque acierte es la precisión salvo si lleva – en cuyo caso siempre acierta.

El daño se calcula de acuerdo a la siguiente fórmula:

$$\text{Daño} = \left(\frac{2xLvl}{5} + 2 \right) x \text{Potencia} x A / D x \text{Stab} x \text{tipo1} x \text{tipo2} x \text{Rnd}$$

Donde A es el tipo de ataque correspondiente a la categoría del movimiento (Mejora 1) y sino el ataque por defecto.

Donde D es el tipo de defensa correspondiente a la categoría del movimiento (Mejora 1) y sino la defensa por defecto.

Stab: (Same type attack bonus) es un multiplicador que se aplica en caso de que el pokemón tenga en uno de sus 2 tipos el mismo tipo que el del movimiento (1,5) y sino toma el valor 1.

Tipo1 depende del tipo primario (primer tipo) del pokemón objetivo (0,5,1 ó 2) y se calcula en función de las debilidades (buscar en internet).

Tipo2 es 1 en caso de que el pokemón objetivo no tenga tipo secundario y sino depende del tipo secundario (primer tipo) del pokemón objetivo (0,5,1 ó 2).

Rnd es un valor aleatorio elegido entre 217 y 255 incluidos y, después, dividido por 255.

Pokeball:

Atributos: ¿?

Métodos: intentarCaptura()

Donde la probabilidad de capturar se calcula de acuerdo con el siguiente algoritmo:

Si el pokemón está bajo algún efecto (Mejora 3) se genera un número entre 0 y 255:

Si el pokemón está congelado y $N < 25$ se captura.

Si el pokemón está paralizado y $N < 12$ se captura.

Sino se logra capturar o no está bajo ningún efecto se calcula f de acuerdo a la siguiente fórmula con la vida del pokemón objetivo:

$$f = \frac{HpActual}{HpMax} \times \frac{255}{3}$$

Se genera un número aleatorio M entre 0 y 255 y se comparan:

Si $M > F$ se logra, y sino fracasa.

Criterios de calificación base:

Criterio	Excelente	Bien	Insuficiente
Presentación visual	Los personajes están debidamente animados y el formato es visualmente atractivo.	Los personajes están debidamente animados	Los personajes no están animados o los divs mal colocados.
Funcionalidad	Las 4 opciones funcionan y el enfrentamiento se desarrolla satisfactoriamente.	Como máximo una de las opciones no funciona, y/o las 4 funcionan pero no se resuelve como debería el enfrentamiento.	3 ó 4 de las opciones del entrenador no funcionan
Legibilidad del código	El código usa nombres descriptivos para las variables y funciones e incluye comentarios acertados en los puntos adecuados (ni en exceso ni en defecto).	El código usa nombres descriptivos para las variables y funciones	El código no usa nombre descriptivos para las variables y funciones

Un alumno que realice el trabajo individualmente y alcance excelente en los criterios anteriores podrá obtener hasta una nota de 9.

Un alumno que realice el trabajo por parejas y alcance excelente en los criterios anteriores podrá obtener hasta una nota de 8.

Un alumno que realice el trabajo en un grupo compuesto por tres personas y alcance excelente en los criterios anteriores podrá obtener hasta una nota de 7.

Además, independientemente de la modalidad, cada ampliación realizada de entre las 4 propuestas sumarán un punto a la nota del trabajo. No es necesario realizarlas en orden (la 1, luego la 2, etc) sino que se pueden añadir bajo criterios de decisión propios.

Se recogen a continuación las 4 posibles ampliaciones:

1ª Añadir los atributos ataque y defensa especial a los atributos de los pokemón y calcular el daño de acuerdo a la categoría del movimiento.

2ª Añadir velocidad a los pokemones y calcular el orden del combate de manera apropiada (más rápido en cada ronda actúa primero).

3ª Añadir los efectos de parálisis y congelación en las habilidades adecuadas y sus efectos en el combate y modificar el algoritmo de captura además de las variaciones en el combate.

4ª Descargar desde PokeApi los sprites mediante un fetch en vez de cargarlos desde un archivo local.

Anexo I: Datos para el trabajo

Atributos de los pokemones que al menos deberán estar programados:

Pokemón	Lvl	Hp	Att	Def	Vel	Att.Esp	Def.Esp
Pikachu	50	142	100	83	138	94	94
Ditto	50	155	61	110	110	61	110
Articuno	50	181	121	136	121	131	161

Movimientos de cada pokemón:

Pikachu	Thunderbolt	Thunder Wave	Agility	Tackle
Potencia	90	-	-	35
Precisión	100%	90%	-	95
PP	15	20	30	35
Categoría	Especial	Estatus	Estatus	Físico
Efecto	10% de probabilidad de paralizar al oponente al impactar	100% de probabilidad de paralizar al oponente al impactar	Aumenta la velocidad del usuario el 100% de su velocidad base	-
Tipo del movimiento	Eléctrico	Eléctrico	Normal	Normal

Ditto	Transformar	-	-	-
Potencia	-	-	-	-
Precisión	-	-	-	-
PP	10	-	-	-
Categoría	Estado	-	-	-
Efecto	El pokemón se transforma en el oponente, copiando los movimientos (que pasan a tener 5 pp cada uno), habilidades y atributos que tenga el otro excepto la salud, que se mantiene.	-	-	-
Tipo del movimiento	Normal			

Articuno	Ice Beam	Agility	Swift	Steel Wing
Potencia	95	-	60	70
Precisión	100%	-	-	90%
PP	10	30	20	25
Categoría	Especial	Estatus	Especial	Físico
Efecto	10% de probabilidad de congelar al oponente al impactar	Aumenta la velocidad del usuario el 100% de su velocidad base	-	10% de probabilidad de aumentar la defensa del usuario un 50% de su defensa base al impactar
Tipo del movimiento	Hielo	Normal	Normal	Acero