

## GUI N DE ACTIVIDAD

# Desarrollo Web en Entorno Cliente

**ACTIVIDAD EVALUABLE:** Persistencia en Cliente y peticiones GET.

### OBJETIVOS

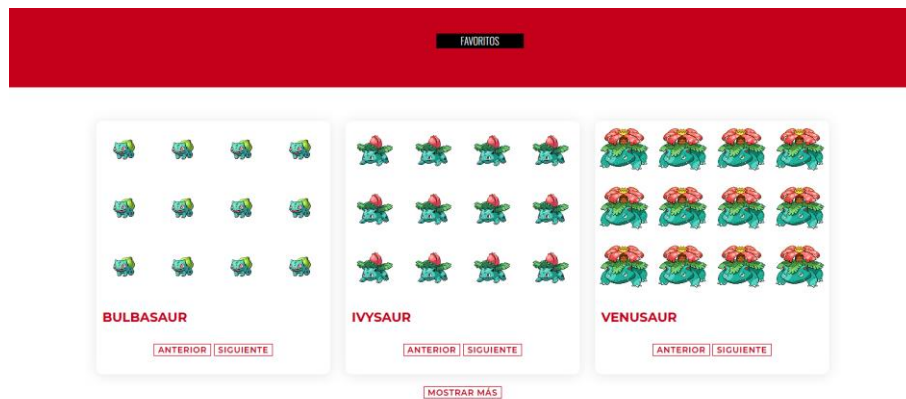
- Realizar peticiones HTTP con JavaScript.
- Guardar datos en el Navegador con JavaScript.
- Implementar funciones de orden superior predefinidas en el lenguaje.

### TEMPORALIZACI N

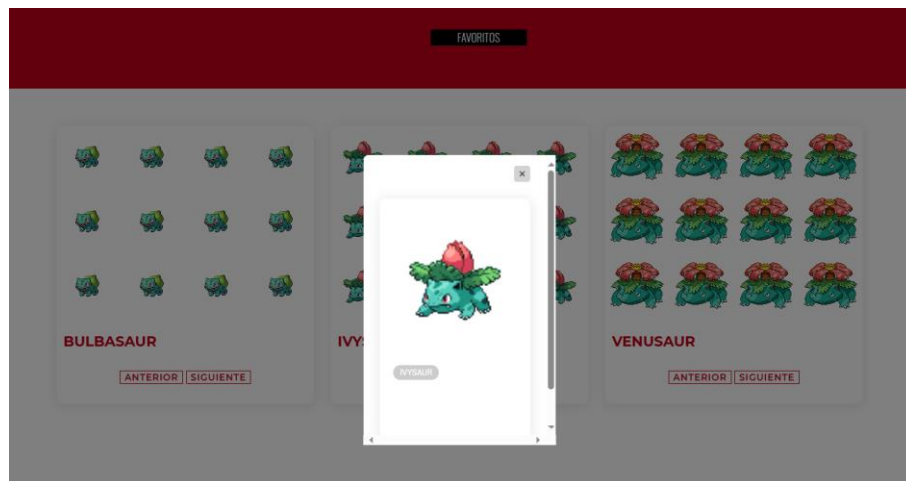
*Aproximadamente 4-5 horas*

### PROCESO DE DESARROLLO

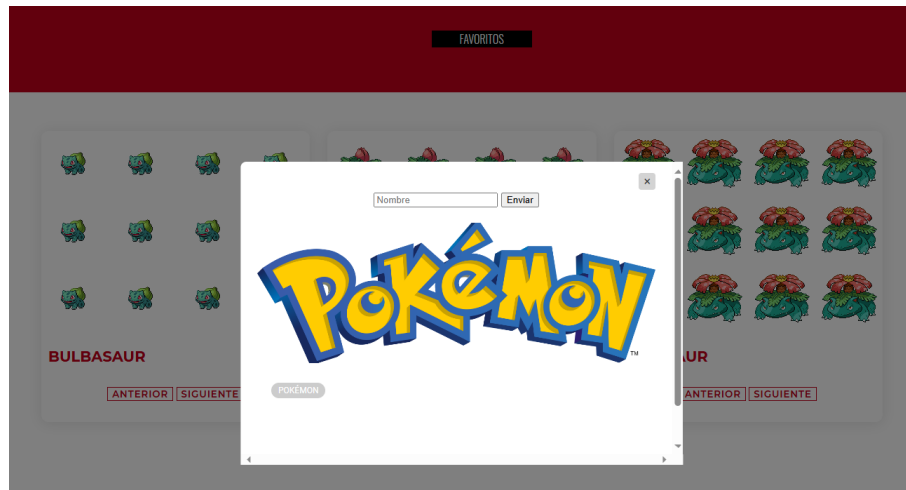
1. Para el archivo index.html y styles.css proporcionados, desarrolla en JavaScript un script main.js con las siguientes funcionalidades:
  - a. Al cargar la p gina se realizar  una petici n Get HTTP a la API REST p blica 'Poke API':
    - [Poke Api](#)
  - b. De la petici n se deber n obtener los siguientes datos: nombre del pok mon y las im genes 'front\_default', 'back\_default', 'back\_shiny' y 'front\_shiny' obtenidas de los 'sprites' de cada Pok mon. Implementa la l gica necesaria para mostrar por pantalla el nombre y una imagen de los tres primeros Pok mon, obtenidos de la primera p gina de la paginaci n que ofrece la propia API. Cada pok mon se deber  mostrar utilizando el div con la clase 'card' proporcionado en el archivo 'index.html':



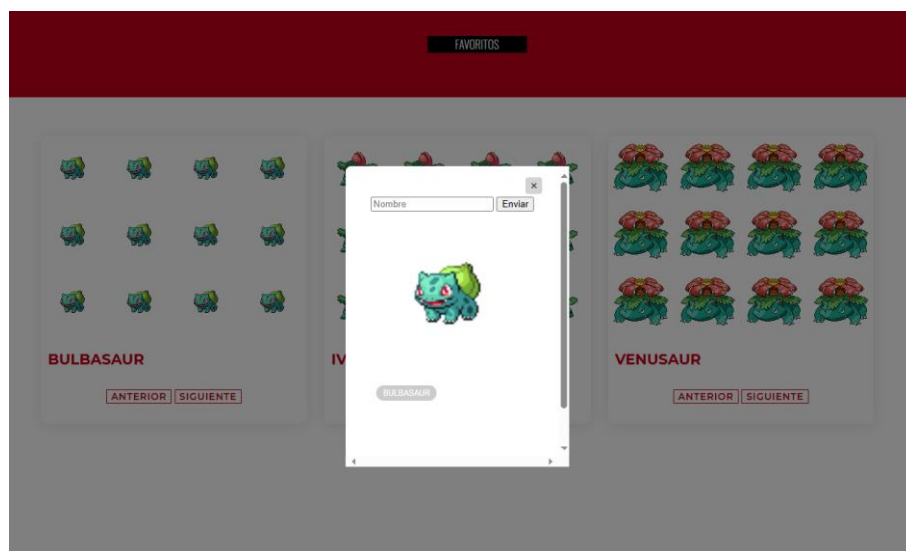
- c. A partir de los botones 'Anterior' y 'Siguiente' de cada *card* se podrá navegar entre las cuatro imágenes obtenidas de cada pokémon –imágenes 'front\_default', 'back\_default', 'back\_shiny' y 'front\_shiny' de la petición Get-, completando un carrousel que permitirá pasar de la primera imagen a la última y viceversa.
- d. Al realizar hover sobre cada uno de los pokémon aparecerá un botón con la inscripción 'Ampliar' –Nota: el hover viene ya implementado mediante CSS, no es necesario realizarlo desde JavaScript-. Al pulsar sobre el botón de un pokémon, aparecerá por pantalla una pantalla modal que permitirá visualizar la imagen de las cuatro que en ese momento se encuentre visible en la página y el nombre del pokémon seleccionado:



- e. Al pulsar sobre el nombre del pokémon de la pantalla modal, éste se añadirá a la lista de favoritos del usuario. Esta lista estará guardada en el almacenamiento local del navegador.
- f. Si se pulsa sobre la inscripción 'Favoritos' ubicada en el Header, se abrirá otra pantalla modal que permitirá realizar una búsqueda de los favoritos guardados en el almacenamiento local del navegador:



- g. Si se introduce un nombre erróneo o un pokémon que no se encuentra almacenado en el navegador, saltará una alerta.
- h. Si se introduce un personaje correcto, se mostrarán sus datos correspondientes en la pantalla modal, tal y cómo se muestra a continuación:



- i. Al pulsar sobre el botón 'Mostrar más' que aparecen debajo de los tres personajes, se borrará este botón y se ampliará el número de personajes hasta totalizar 20 pokémon. Se crearán dos botones con las inscripciones 'anteriores' y 'siguientes', que permitirán navegar por las páginas de la paginación proporcionada por la API. De la primera página se deberá de poder acceder a la última y viceversa.
- j. En cada una de las páginas, se mostrarán por defecto inicialmente tres pokémon y al pulsar sobre el botón 'Mostrar más', se ampliarán hasta mostrar los 20 totales que ofrece la paginación de la API.
- k. Para esta actividad no se permitirá la utilización del bucle for, penalizándose su uso.

## EVALUACIÓN

---

Esta actividad corresponde a la parte de actividades evaluables de la evaluación.

## OBSERVACIONES

---

Se entregará un único archivo comprimido con el archivo main.js. La entrega se realizará en el espacio habilitado para ello en Florida Oberta. **El script a entregar deberá ceñirse a lo que se pide explícitamente en el enunciado de la actividad. Cualquier desviación de las funcionalidades solicitadas o la implementación de funcionalidades no solicitadas de manera explícita en el enunciado, conllevará una penalización sobre la nota total.**