

Teamwork

 OPEN REPOSITORY




SEND TO REVIEW

MORE OPTIONS 



















In progress

Team

Project Team



Ratings (96)

DIFFICULTY:		3.9
USEFULNESS:	     	4.4
FUN:	     	4.4
MATERIALS:	     	3.7

Gotta fetch() 'Em All

In this task, you will be tasked with creating the coolest Pokemon site possible! With [PokéApi](#), you have access to all existing Pokémon and their data, create a great application based on the description, but feel free to expand this great project with new ideas!

What are you going to learn?

- Handling API request and responses

Tasks

SHOW ALL CRITERIA

0%

☐ Locations

Show the [received locations](#) (first 20) on the site.

1. The first 20 locations are present on the site.

CRITERIA 

☐ Let's find some Pokemons

When the user clicks on a location, the game finds a random pokémon in that location and an encounter starts.

1. The locations disappear when clicking on a location.
2. The name and sprite (front_default) of the encountered pokémon appears on the page.
3. If the selected location doesn't have any pokémon encounter, then show the message "This location doesn't seem to have any pokémon" and a button. When clicking on the button, the list of locations should be displayed again.

CRITERIA 

☐ The battle begins

When the encounter begins, pick one of your own pokémon to fight. Since we don't have a Catch function in our code, right now you just need to hard code the chosen pokemons into a list eg.:

```
// starting pokemons
const usersPokemon = [
  "https://pokeapi.co/api/v2/pokemon/bulbasaur",
  "https://pokeapi.co/api/v2/pokemon/charizard",
  "https://pokeapi.co/api/v2/pokemon/poliwhirl"
]
```



1. The name and sprite of all pokémon owned by the player are shown at the beginning of the encounter.
2. The user can pick one of their own pokémon and the selected pokémon is displayed as well as the encountered one.

CRITERIA ^

☐ **The encounter**

The player and the encountered (opponent) pokémon take turns attacking each other until the hp of one of them reaches 0.

1. When a pokémon takes its turn, some damage is subtracted from the other pokémon's hp. The damage is calculated using the following formula:
 $((((2/5+2) * B * 60 / D) / 50) + 2) * Z / 255$, where B is the attacker's Attack, D is defender's Defense, and Z is a random number between 217 and 255. The HP, Attack, and Defense can be read from the API at `pokemon.stats` (see for example: `pokemon.stats[0].stat.name`).
2. When the encountered pokémon's hp reaches 0 while our pokémon still has positive hp, then the encountered pokémon is captured, and it is added to the player's list of pokémon.
3. When our pokémon's hp reaches 0 first, then the encounter ends.
4. After the encounter, the available locations is shown again for the user to go somewhere else.

CRITERIA ^



Hints

- Getting all Pokémon on location:

```
location.areas[rand].url ->
let area = fetch("https://pokeapi.co/api/v2/location-
area/1/")
area.pokemon_encounters[rand].pokemon.url
```

- The sprite can be scaled up and preserve its pixelated style by using the following style on the `img` element: `image-rendering: pixelated;`

Background materials

[PokéApi](#)
[Complete Battle Damage formula](#)