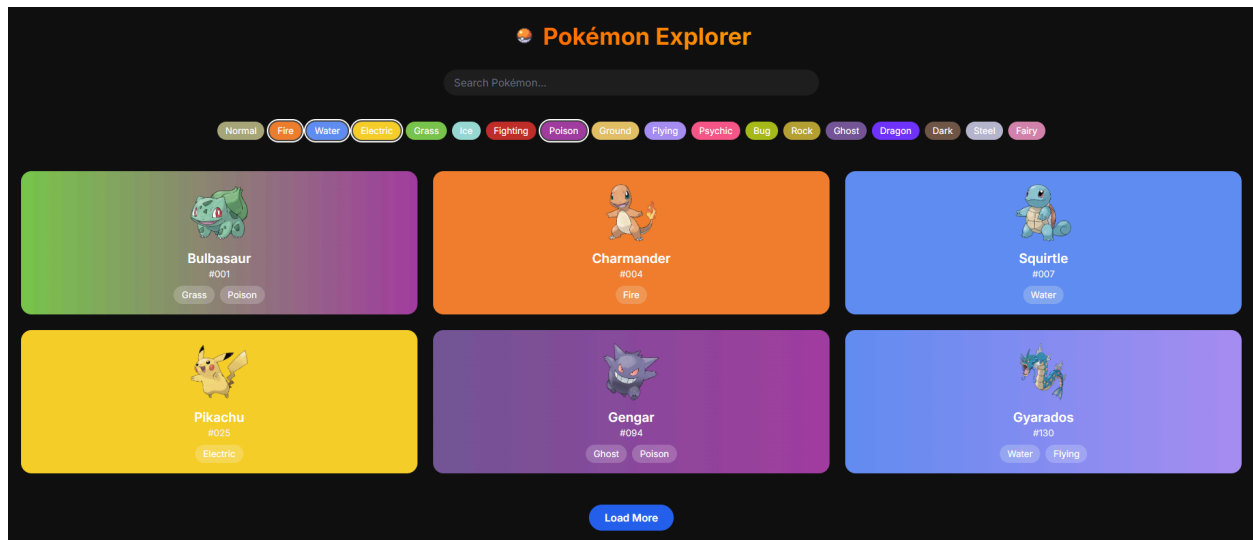


Wave

Frontend Dev 2025 - PokeApi



Opis projektu

Twoim zadaniem będzie wykonanie prostej aplikacji internetowej, która umożliwiać będzie przeglądanie pokemonów wg. ich typów. Z naszej strony dostaniesz namiastkę designu w formie PNG, którego implementację wykonasz w ramach zadania. Im bardziej Twoja implementacja będzie go przypominać, tym lepiej ocenimy wykonanie zadania.

Technologie

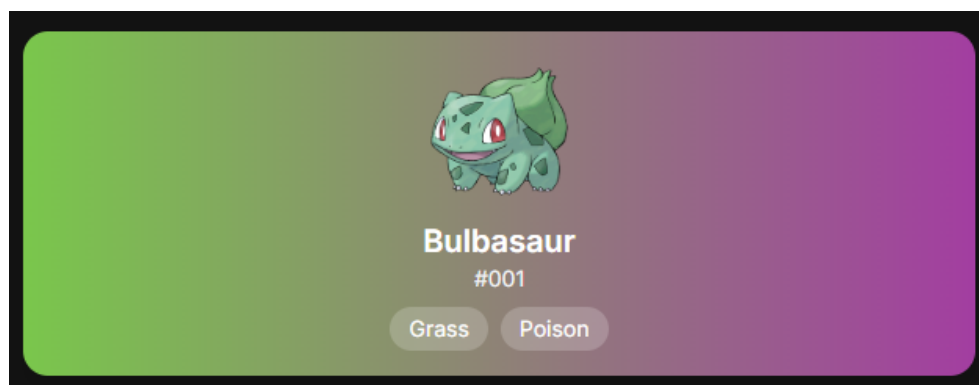
Wykonane zadanie powinno:

- wykorzystywać bibliotekę komponentów (MUI, ArkUI, Shadcn lub inną)
- mieć konsekwentne podejście do stylowania (css modules, tailwind, css-in-js, etc.)
- wykorzystywać React lub Vue

Do pobierania danych z API wykorzystaj “fetch”. Może on być opakowany w bibliotekę do zarządzania stanem żądania, np. TanStack Query

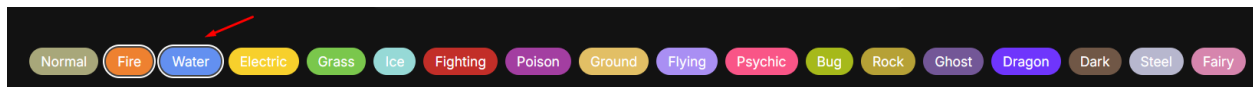
Wymagania

1. Lista pokemonów w formie kart. Każda karta musi zawierać:
 - a. Miniaturkę pokémona
 - b. Nazwę
 - c. Numer porządkowy
 - d. “Pille” z typami
 - e. Kolorowe tło odpowiadające typom (w przypadku więcej niż jednego typu, gradient płynny). **PokeApi nie udostępnia listy kolorów** - znajdziesz je w sekcji “Materiały i zasoby” na dole tego dokumentu.
 - f. Dodatkowe punkty za implementację dynamicznego doładowywania pokémonów (przycisk “Load more” lub scroll/infinite scroll, max 24 na “stronę”)



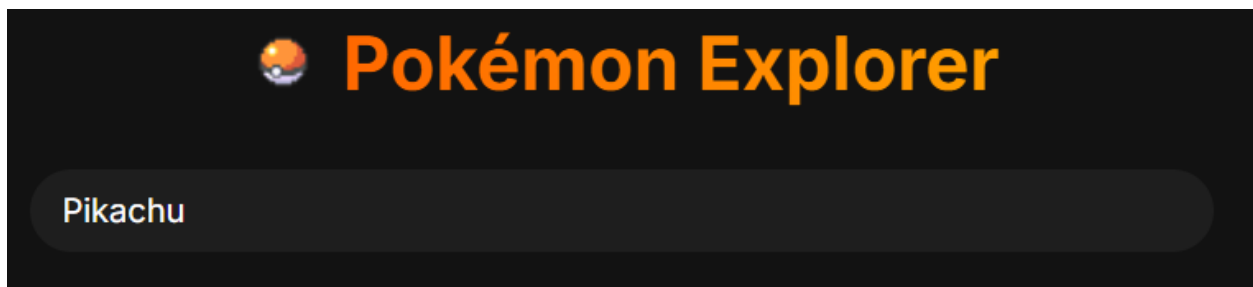
2. Filtry typów

- Każdy typ powinien być wyświetlony w formie pigułki (Pill) nad listą pokemonów
- Po kliknięciu na dowolnego Pill'a, musi nastąpić filtrowanie pokemonów danego typu/typów.
- Możliwe jest wybranie wielu typów jednocześnie.
- Zaznaczone filtry muszą posiadać jasną obwódkę



3. Nagłówek i search

- Tytuł - Pokemon Explorer
- Ikona pokeballa obok nazwy strony
- Search bar - filtrowanie/wyszukiwanie automatyczne po zakończeniu pisania



- Modal ze szczegółami pokemona (po kliknięciu na element listy), galerię jego sprite'ów/obrazków, ewolucjami oraz listą lokacji, gdzie można go znaleźć. Wygląd wedle uznania, ale wykorzystaj dostępne komponenty z wybranej biblioteki.

Nice to have, czyli za co dostaniesz dodatkowe punkty

- Responsywność
- Semantic HTML + Aria
- Interaktywność elementów (hover, active, itp)
- Dodawanie pokemonów do ulubionych oraz podstrona z ich listą (zapisz dane w localStorage)
- Podzielenie build'u aplikacji na dev oraz prod, z odpowiednimi optymalizacjami

Ocena kodu/aplikacji

W ramach oceniania będziemy patrzeć zarówno na kod jak i na działanie aplikacji. Im bardziej czytelny i uporządkowany kod działającej aplikacji dostarczysz, tym lepiej. Podobnie z odwzorowaniem wyglądu apki. Wybór technologii nie ma znaczenia, ale pamiętaj że **korzystanie z AI jest zabronione i powoduje automatyczną dyskwalifikację!**

Żeby ocenić Twoją pracę, będziemy od Ciebie potrzebować kodu, instrukcji uruchomienia programu oraz ewentualnie link do działającej aplikacji uruchomionej na dowolnie wybranym przez Ciebie serwisie (np. GitHub Pages, Vercel, Netlify, itp). Dodatkowo, w pliku **readme.md** powinien znajdować się opis/dokumentacja aplikacji.

Materiały i zasoby

1. PokeApi
 - a. <https://pokeapi.co/>
2. Pokeball
 - a. <https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/items/poke-ball.png>
3. Design - desktop
 - a. https://testy.wavedc.pl/zadania/wave_desktop.png
4. Kolory typów
 - a. <https://pastebin.com/Eun1VvHH>

Powodzenia!