

# I. Présentation du projet

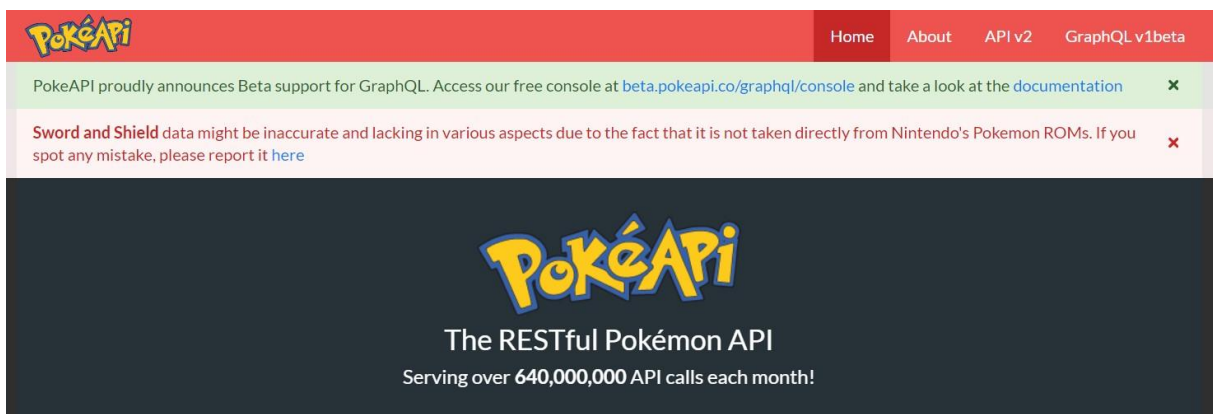
**Contexte** : Le projet "PokéAPI" est une application web développée dans le cadre de la première année du BTS Services Informatiques aux Organisations (SIO). Ce projet a été conçu pour démontrer la capacité à interagir avec une API externe et à présenter des données de manière dynamique et interactive.

**Objectif** : L'application vise à afficher une liste de Pokémon, en tirant parti des données fournies par la PokeAPI, une source externe riche en informations sur les Pokémon. Elle permet aux utilisateurs de parcourir les Pokémon, visualiser des détails basiques, et naviguer entre différentes pages de résultats.

**Technologie** : Le choix de React pour ce projet est stratégique, compte tenu de sa popularité et de son efficacité dans la création d'applications web modernes. React aide à construire une interface utilisateur dynamique et réactive, gérant efficacement l'état et le rendu des composants.

## II. Utilisation d'une API

L'application s'appuie sur **PokeAPI**, une interface de programmation d'applications (API) publique qui fournit des données complètes sur les Pokémon. Elle offre des informations détaillées, comme les types de Pokémon, leurs statistiques, et leurs images.

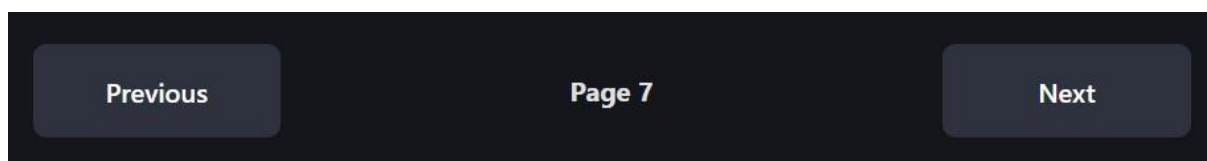


Pour interroger l'API, le projet utilise **Axios**, un client HTTP basé sur les promesses. Axios simplifie le processus d'envoi de requêtes HTTP asynchrones, permettant de gérer les réponses de l'API de manière plus lisible et maintenable.

```
setLoading(true)
let newPokemons = [];
let pageItemsNumber = 10;
axios.get(`https://pokeapi.co/api/v2/pokemon?limit=${pageItemsNumber}&offset=${page * pageItemsNumber - pageItemsNumber}`)
  .then(async response => {
    let promises = [];
    for (let pokemon of response.data.results) {
      promises.push(axios.get(pokemon.url).then(response => {
        newPokemons.push(response.data)
      }));
    }
    await Promise.all(promises);
    setPokemons(newPokemons)
    setLoading(false)
  })
  .catch(error => {
    console.log(error);
  });
```

## III Gestion des pages

Un aspect essentiel de l'application est la gestion de la **pagination**. La pagination permet aux utilisateurs de naviguer à travers de grandes quantités de données sans surcharger l'interface utilisateur.



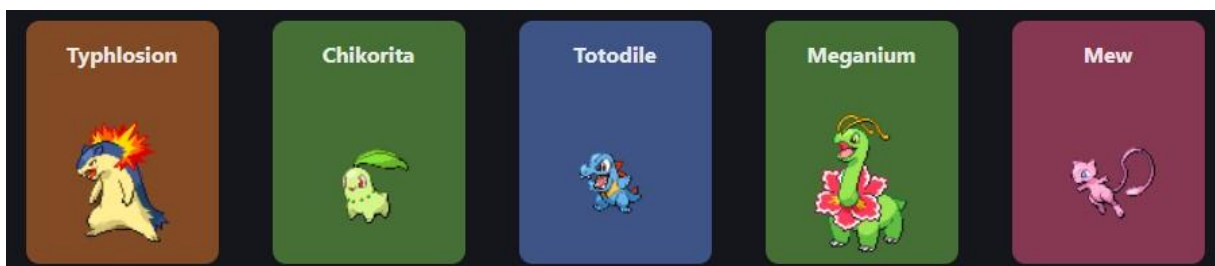
L'utilisation des **hooks** d'état de React (useState) permet de gérer le numéro de la page actuelle (page) et l'état de chargement des données (loading). Cette approche rend l'application plus réactive et améliore l'expérience utilisateur.

```
const [page, setPage] = useState(1);
const [loading, setLoading] = useState(false);

<div className='pages'>
  <button onClick={() => setPage(page - 1)} disabled={page === 1}>Previous</button>
  <p>{loading ? 'Loading...' : `Page ${page}`}</p>
  <button onClick={() => setPage(page + 1)}>Next</button>
</div>
```

## IV Affichage des pokémons

Chaque Pokémon est présenté avec son nom et une image. Un style dynamique est appliqué pour refléter le type principal de chaque Pokémon.



La fonction **colorFromPokemonType** est un élément clé qui assigne des couleurs spécifiques aux différents types de Pokémon, améliorant ainsi l'aspect visuel de l'application.

```
function colorFromPokemonType(type) {  
  switch (type) {  
    case 'normal':  
      return '168, 167, 122';  
  }  
}  
  
<div key={index} style={{ backgroundColor: `rgba(${colorFromPokemonType(pokemon.types[0].type.name)}, 0.5)` }}>  
  <h2>{pokemon.name[0].toUpperCase() + pokemon.name.slice(1)}</h2>  
  <img src={pokemon.sprites.front_default} alt={pokemon.name} />  
</div>
```

## IV Conclusion

Le projet "**PokéAPI**" démontre une application réussie des concepts clés du développement frontend, notamment **l'utilisation de React, la gestion de l'état, l'interaction avec les API externes via Axios**, et la présentation dynamique des données. Cette application sert non seulement de démonstration pratique des compétences acquises dans le cadre du BTS SIO, mais elle offre également une expérience utilisateur attrayante et informative pour les fans de Pokémon.

Perspectives : Ce projet pourrait être étendu pour inclure des fonctionnalités supplémentaires, telles que la recherche de Pokémon par nom, l'affichage de détails supplémentaires sur chaque Pokémon, ou même l'intégration de fonctionnalités interactives comme la création d'un