



**420-211**  
**Applications Web**

**TP1**  
**Applications basées sur les données**

**Prénom & Nom :**

**Matricule :**

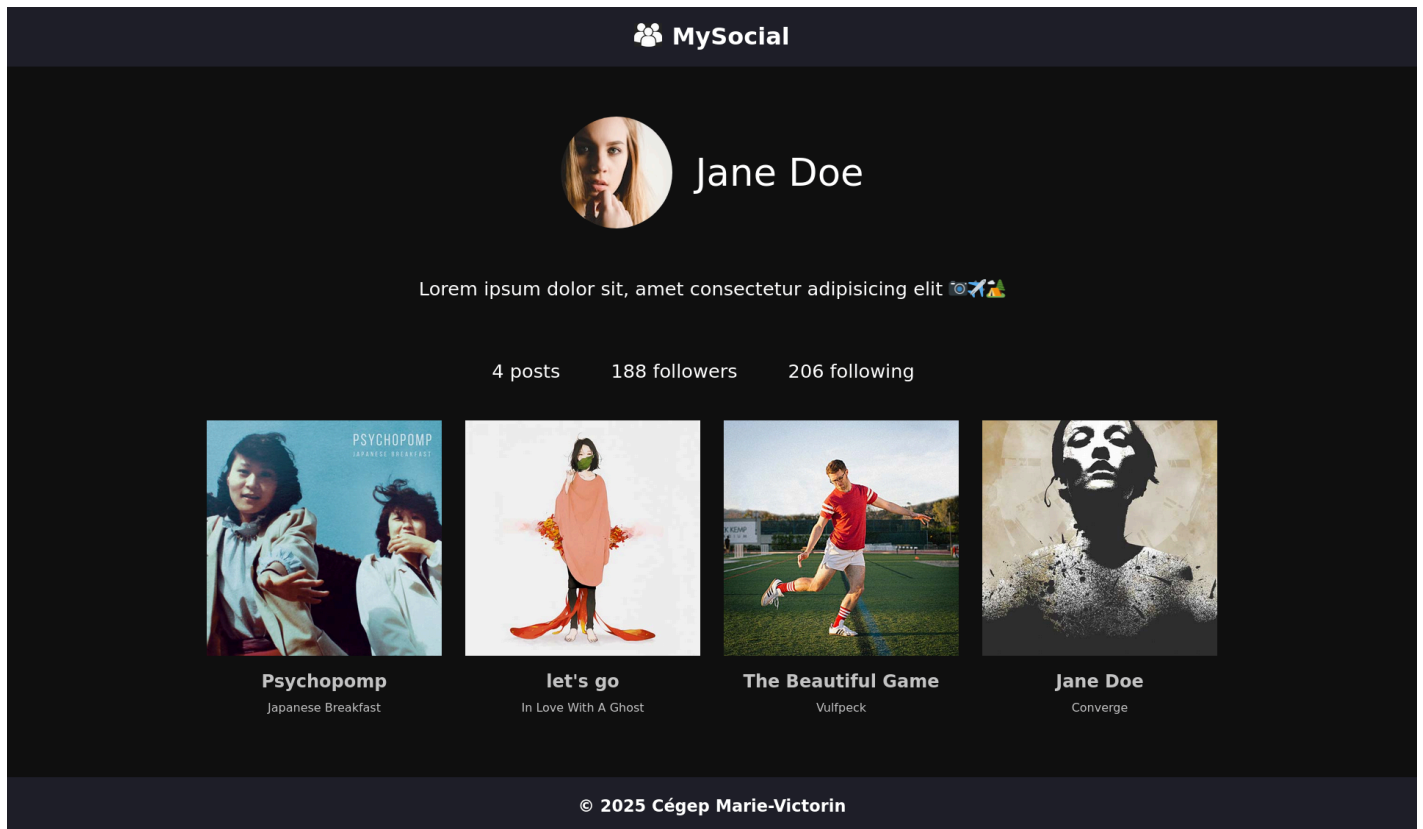
**Signature :**

## Objectif

Vous devez faire une application React simple utilisant le passage de « props ». L'application affiche le profil d'une personne.

## Spécifications

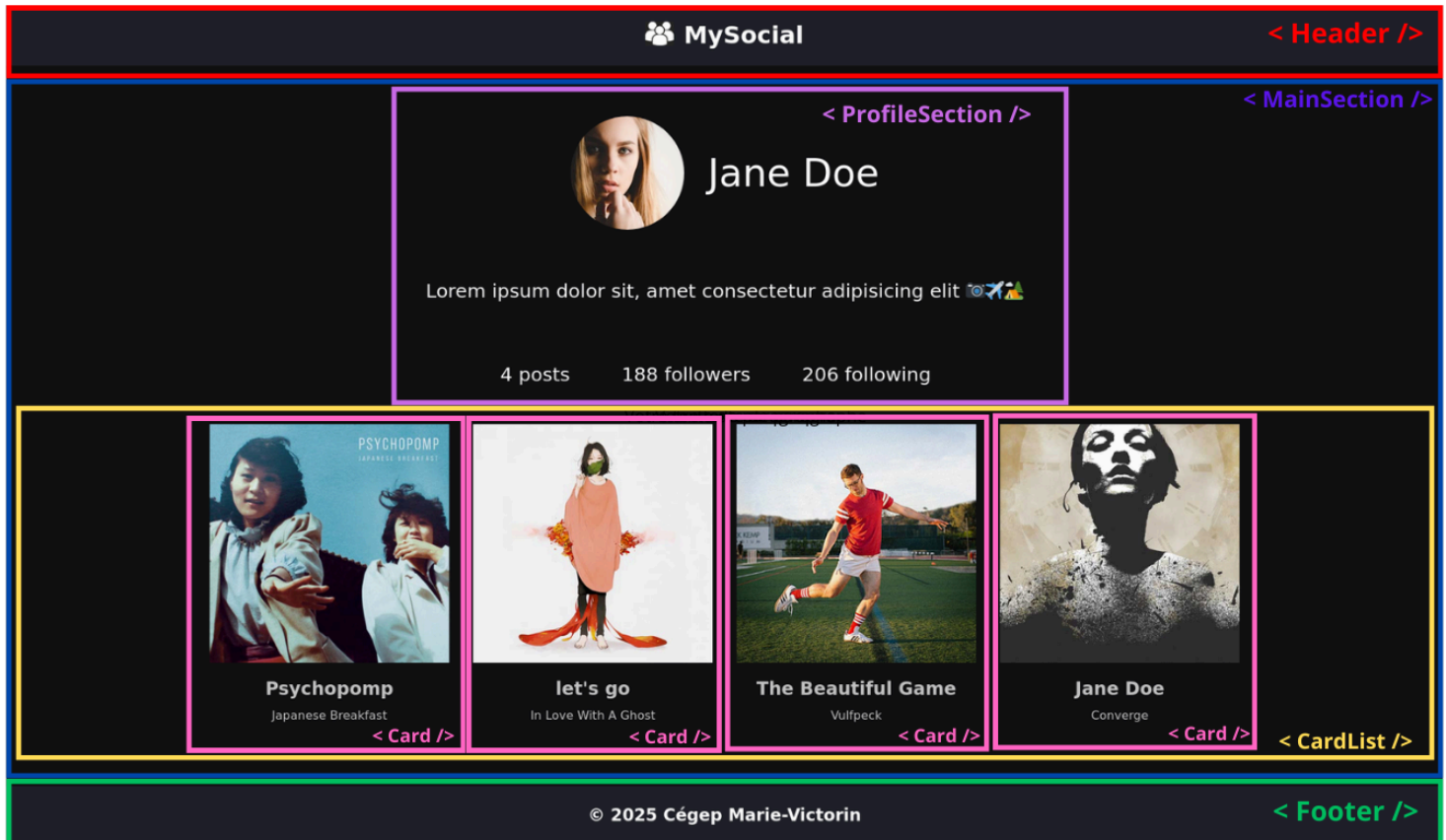
La page doit avoir la disposition suivante :



Les composantes React à créer sont les suivants :

```
< Header />
< MainSection />
  < ProfileSection />
  < CardList />
    < Card />
< Footer />
```

Ces composants divisent la page comme suit :



- Vous pouvez créer d'autres composants au besoin.
- Les valeurs affichées dans la page doivent correspondre aux propriétés d'un objet JavaScript.
- Un modèle de la page en HTML ainsi qu'un fichier de style CSS vous sont fournis dans le répertoire modèles (profil.html et profil.css).

## Consignes

- Votre projet React doit contenir les fichiers suivants :
  - App.jsx (entrée de l'application),
  - Header.jsx,
  - MainSection.jsx,
  - ProfileSection.jsx,
  - CardList.jsx,
  - Card.jsx,
  - Footer.jsx

- Les données liées au profil (photo de profil, nom, prénom, description, statistiques de nombre de posts, d'abonnés et d'abonnement, les images, titres et leur description) doivent être toutes stockées dans un ou des objet(s) Javascript. Vous pouvez choisir le nom, prénom, l'image du profil, la description ainsi que les photos et description des photos que vous voulez (vous pouvez aussi garder celles fournies), tant que vous utilisez des objets javascript pour les déclarer.
- Les données doivent être passées aux composants **via des props**.
- Les *Cards* doivent être dynamiquement générées compte-tenu d'un tableau décrivant ces cards. Vous devez utiliser `.map()` pour les générer.
- Pour la statistique du nombre de posts, celle-ci doit être **dynamiquement égale au nombre de posts du profil**.
- Enfin, divisez le fichier de style `profil.css`. Pour chaque composant, créez-lui son propre fichier de style (i.e. `<NomDeLaComposante>.css`).

## Consignes de remise

- Vous devez publier votre code sur GitHub en utilisant le dépôt (*repo*) fourni par l'enseignant.
- Votre travail doit être remis dans **une branche nommée TP1**
- **Date limite de remise : le jeudi 6 mars avant minuit.**

## Grille de correction

Efficacité du code <ul style="list-style-type: none"> <li>• Pas de répétitions</li> <li>• Bonne utilisation d'objet(s) JavaScript décrivant les propriétés du profil</li> <li>• Bonne séparation de la page en composants réutilisables</li> </ul>	/4
Fonctionnalité du code <ul style="list-style-type: none"> <li>• Utilisation des props pour faire passer les propriétés aux composants</li> <li>• Utilisation de <code>.map()</code> pour dynamiquement créer les composants Card</li> <li>• Statistique du nombre de posts correspond au nombre de posts (de façon dynamique)</li> </ul>	/4
Disposition des éléments conforme aux exemples	/2
Pénalités pour non respect des consignes de remise (retard, mauvaise branche GitHub ou autre)	
<b>TOTAL</b>	<b>/10</b>