

Syrine Radhouane

Projet: <https://github.com/TiFosi/prog-web>

Groupe AR

Identifiant github: SyrineRadhouane588

Tâches effectuées:

- En se basant sur le composant Table du framework AntD, j'ai implémenté le composant DataTable qui affiche un tableau de données triées par jour avec la possibilité de trier sur les colonnes.
- J'ai implémenté côté backend la route qui permet de récupérer de la base de données mongodb les taux d'incidence sur toute la france.
- Intégrer les données récupérées du backend dans le composant DataTable.
- Modifier les composants App, DataTable et Chart afin que la récupération des données se fasse au niveau App parce que les mêmes données sont consommées par les deux composants DataTable et Chart.
- J'ai implémenté la partie front du composant Filters en se basant sur le framework AntD.
- J'ai implémenté les routes côté back pour récupérer de la base de données les listes de régions et de département et lier le composant Filters par ces données.
- Refactorer le composant App parce qu'il est devenu gros et mettre la partie Content dans un composant Main.

Stratégie employée pour la gestion des versions avec Git:

- Au début du projet on s'est mis d'accord à aborder la stratégie suivante afin de contribuer au code:
 - On a défini deux branches principales protégées: master et develop
 - Les développements se font en local dans des branches qui ont des noms significatifs par rapport au contenu
 - Après le push de la branche, on ouvre une pull request pour merger le contenu de la branche dans develop. Cette pull request doit être approuvée par un autre afin que les modifications soient prises en compte
 - La branche master est faite pour merger des versions stables et testées de notre code

Solutions choisies:

- On a choisi d'utiliser AntD comme framework UI pour sa simplicité et sa documentation très détaillée. On a utilisé Bootstrap au sans composants react début, mais on a vite changé vers AntD qui propose directement des composants React

- Côté Backend on a utilisé express.js comme framework, et une base de données NoSql MongoDB sur le cloud (MongoDB Atlas). On aurait pu utiliser une base de données en local mais avec le cloud c'est plus simple.

Difficultés rencontrées:

- Dans le démarrage du projet, le besoin n'étant pas clair, on a voulu utiliser plusieurs datasets du site santé Santé public France mais on ne savait pas comment faire d'une façon rapide l'intégration de ces datasets dans notre base de données mongodb. Alors, on a beaucoup cherché jusqu'à trouver l'outil mongoimport qui nous a permis d'importer plusieurs datasets dans la BD cloud.
- On a rencontré un problème avec l'affichage du composant chart. Par défaut l'affichage ne se fait pas exactement sur la largeur du conteneur sauf si on agrandit ou on réduit la fenêtre. on n'a pas pu corriger ce comportement, même avec les react-columns qui affiche la Chart sur une largeur plus grande que le conteneur.

Un composant que j'ai écrit et qui me semble élégant:

Le composant DataTable. Ce composant récupère du backend les données sur le taux d'incidence standardisé et les affiche dans un tableau. J'ai utilisé le composant Table de AntD qui prend deux arguments: les colonnes et les données. Pour afficher les données dans un ordre descendant j'ai trié les données dès la partie back. J'ai formaté les informations affichées selon leurs types: Date "mardi 23 février 2021" et les nombres flottants. Dans un second temps, j'ai fait sortir la partie récupération de données de ce composant et l'ai mis dans Apps parce que le composant Chart aussi consomme les mêmes données. Donc toute la partie useEffect est gérée dans un composant parent et ce composant juste prend les données en props.

Un composant que j'ai écrit et qui mériterait une optimisation ou amélioration:

Le composant Filters: Ce composant gère les routes dans notre application. Il y a trois routes: accueil, par région et par département. Dans l'accueil, par défaut les données chargées dans les composants DataTable et Chart sont par rapport à toute la France. Dans ce contexte, j'ai lié les trois routes sur trois événements: onClick bouton France, onChange menu Par région et onChange par département. Je remplis les menus déroulants de régions et départements du Backend. Ce qui mérite l'amélioration dans ce composant c'est la cohérence des boutons et les événements. Je veux que par défaut les 3 boutons soient affichés sans menu à côté, juste au moment du clic le menu correspondant s'affiche et au moment du select tout le groupe devient actif et déclenche la mise à jour des autres composants par les données chargées. Une partie de cela est implémentée mais n'est pas optimale, déjà on observe le chargement du Chart plusieurs fois au moment de sélectionner un élément de la liste.