

Devoir (noté) n°4 : Application web avec React

Projet React - Portail Admin du portail Wikilogie

Contexte

Wikilogie est un portail développé par des étudiants du GI de l'UTC. Il est géré par l'association du BDE Wikilogie UTC. Sa vocation est de promouvoir les principes de l'ingénierie soutenable en proposant des articles, des bonnes pratiques, des quizz, etc, autour du sujet. Ces ressources proviennent directement de la communauté que ce portail abrite (les utilisateurs inscrits).

Pour bien fonctionner, le portail repose sur une base de données relationnelle et un index de fichier. Il faut donc que les développeurs et les administrateurs du fichier puissent avoir à tout moment une vue globale de l'ensemble des données disponibles.

Fonctionnalités demandées

Lorsque l'on se connecte au portail, on doit :

- consulter les tables de la base de données (en permettant aussi de sélectionner les colonnes à afficher),
- modifier/supprimer des entrées de la base de données,
- sur une table, faire une requête en renseignant des critères sur les champs de la table (ex: "login = 'toto'", "age < 20"),
- **bonus** : consulter le contenu d'un index ElasticSearch,
- **bonus** : effacer tout le contenu d'un index ElasticSearch,
- **bonus** : peupler un index à partir d'un fichier json.

La plupart des opérations pourront se faire via des formulaires.

Vous veillerez à respecter au mieux les bonnes pratiques en matière de sécurité.

Technologies

Il faut créer une application web à partir des technologies ReactJS (client) et NodeJS (server). L'application devra communiquer avec une base de données PostgreSQL et une instance Elastic Search (bonus).

Ressources

Un projet GitLab est mis à disposition à l'adresse suivante :

https://gitlab.utc.fr/wikilogie_dev_team/portail-admin-wikilogie.git

(https://gitlab.utc.fr/wikilogie_dev_team/portail-admin-wikilogie.git)

Il faut créer un 'fork' de ce projet à partir de la branche **develop** vers son espace personnel GitLab.

Attention : **il est interdit de push ou de merge directement dans ce projet**. Afin de restituer votre proposition, vous devrez ouvrir une merge request vers la branche develop du projet initial.

Sur ce projet, un squelette de l'application vous est imposé et des packages sont pré-installés ou suggérés (cf. section Packages à utiliser (<https://pad.rhizome-fai.net/vu7GRwq8RhmfvcJeulSI1A?both#Packages-%C3%A0-utiliser->)).

Vous trouverez :

- un dossier *client* :
- un dossier *server* :
 - un dossier *routes* : contient les fichiers où sont décrits les routes,
 - un dossier *sql* : contient les scripts et dossier utiles pour la connexion et le requêtage de la base de données,
 - un fichier **index.js** : point d'entrée du serveur. On y déclare les routes,
 - des fichiers **package-lock.json** et **package.json** contenant les dépendances et les informations sur l'application.

Dans le dossier *server/sql*, vous trouverez un fichier **init.sql** qui vous permettra de créer et d'initialiser la base de données PostgreSQL sur votre machine.

Pour l'exécuter à partir du client psql, vous pouvez accéder à l'emplacement du fichier (`cd server/sql/`) et lancer la commande suivante : `psql -h localhost -U postgres -p 5432 -a -q -f init.sql`

Commandes utiles :

- lancer le client : `npm start`
- installer les packages (dans chacun des dossiers *client/* et *server/*) : `npm i`
- lancer le serveur : `node index.js`

Packages à utiliser :


Nom	Type	Installé?	Utilité
axios	client	oui	client HTTP pour requêter la base de données
pg	serveur	oui	gestion de la communication entre le serveur et PostgreSQL
nodedemon	serveur	oui	ce package permet de redémarrer automatiquement le serveur à chaque modification du code source du serveur

Nommages des entités

Il faut veiller à respecter les règles de nommages des fichiers, variables et fonctions.

- tous les noms doivent être en anglais et le plus explicites possibles,
- **fichiers** : en minuscule et en snake case (*my_file.extension*)
- **variables et paramètres** : en minuscule et camel case (*myVariable*)
- **fonctions** : en minuscule et camel case, doit commencer par un verbe (*findUser()*)

Livrables et date limite

- **Date limite** : 24/06/2021
- **Nombre de personnes autorisées sur le même groupe** : "3-4"
- Vous devez déposer sur Moodle un seul fichier text (NomsEtudiants.txt) qui contient le lien gitlab de votre dépôt devoir REACT. Ce dépôt contient deux livrables :
- **Code sources**
- **Rapports** :
 - un fichier NomsBinomesDevoir4.pdf avec deux sections :
 - Explications détaillées de la mise en œuvre de l'application, les considérations écoresponsables, résultat d'analyse greenIT de votre application, problèmes rencontrés, etc.! 
 - Screenshots de votre application.
 - Un fichier PDF listant les éventuels packages ajoutés et leur utilité.
- Une merge request sur le projet Portail Admin Wikilogie.

Contacts et aide :

- Ahmed Lounis ahmed.lounis@hds.utc.fr (mailto:ahmed.lounis@hds.utc.fr)

- Mernissi, Aida El aida.el-mernissi@etu.utc.fr (<mailto:aida.el-mernissi@etu.utc.fr>)
- Moravsky, Vladimir Martin Tchekovy vladimir-martin.tchekovy-moravsky@etu.utc.fr (<mailto:vladimir-martin.tchekovy-moravsky@etu.utc.fr>)