

Consignes :

- Le TP est un sujet à faire sur Vue 3, en utilisant TypeScript, Vuex, et Vuetify pour le design system.
- Les bonnes pratiques de développement sont attendues (des documentations sont disponibles sur le Wiki NoConsulting) :
 - Respect des bonnes pratiques Git
 - Utilisation du Camel Case
 - Documentation
 - ...
- Livraisons attendues :
 - Documentation technique
 - Pull Request régulières via le Gitlab de NoConsulting

Sujet du TP :

Votre client vous demande de créer une « bièrothèque ». Pour ce faire, il vous a fourni une liste d'exigences, ainsi que des demandes bien précises. Attention, l'idée est de fonctionner en mode projet. Le client n'a peut-être pas correctement listé toutes ses exigences, et n'a peut-être pas pensé à toutes les contraintes, besoins, nécessaires au bon développement et au bon fonctionnement de l'application. Soyez force de proposition.

Le développement du back-office, du serveur de web-socket, ainsi que les spécifications techniques – et notamment les contrats d'interface – ont déjà été effectué par d'autres développeurs ultra gentils. Si jamais elles ne sont pas suffisantes, les développeurs de ces applications sont disponibles et listés dans la partie consigne.

Attendu client :

Une application web permettant de lister, d'ajouter, de modifier, de supprimer des bières. L'application doit être responsive. Le client vous fait entièrement confiance pour la partie graphique de l'application, même si une validation de celle-ci doit être prévue avec lui.

Spécifications fonctionnelles :

Ajout d'une bière :

Une page dédiée doit permettre à l'utilisateur d'ajouter une bière. La description d'une bière est indiquée dans les spécifications techniques. Les règles de validation du formulaire seront déduites grâce à l'objet bière présent dans les spécifications techniques.

Il est demandé également d'afficher de façon automatique, le prix TTC (TVA à 20%) d'une bière lors de l'ajout du prix HT dans le formulaire. Ce prix TTC ne doit pas être enregistré en BDD.

Lors de l'ajout d'une bière, si celui-ci, et uniquement si, s'est correctement déroulé, l'utilisateur sera redirigé sur la liste des bières.

Liste des bières :

Une page dédiée doit permettre à l'utilisateur de lister toutes les bières présentes dans sa collection de bières. Cette page sera la page d'accueil de l'application.

La liste de bières affichera pour chaque bière les caractéristiques suivantes :

- Le nom de la bière : l'affichage de ce champ doit être normalisé pour être affiché uniquement en majuscule
- Le prix TTC
- Le commentaire

- Le degré d'alcool

Cette liste doit pouvoir être triée par :

- Ordre alphabétique sur le nom (ascendant et descendant)
- Prix TTC (ascendant et descendant)

Une recherche sur tous les critères d'une bière doit pouvoir être également effectuée par l'utilisateur pour lui permettre de retrouver facilement une bière.

Sur chaque ligne de bières plusieurs actions sont disponibles via un bouton :

- La modification : permet d'accéder à un formulaire de modification de la bière (voir partie sur la modification d'une bière).
- La suppression : permet de supprimer définitivement une bière après avoir confirmé la suppression via une modal (voir partie sur la suppression d'une bière).

Attention, le client a également exigé une pagination de la liste des bières.

Modification d'une bière :

L'utilisateur, depuis la liste des bières, doit pouvoir modifier chacune des bières de la liste. Il doit pouvoir éditer tous les champs de la bière comme lors d'une création.

Lors de la modification d'une bière, l'utilisateur sera redirigé sur la liste des bières, uniquement si la modification s'est correctement effectuée.

Suppression d'une bière :

L'utilisateur, depuis la liste des bières, doit pouvoir supprimer chacune des bières de la liste. Au clic sur le bouton de suppression, une modal de confirmation lui demande d'approuver la suppression de la bière.

Mise à jour automatique du prix d'une bière :

Le prix des bières étant indexés sur la bourse de la cuite et le prix des céréales, ceux-ci fluctuent. Ainsi, les prix doivent se mettre automatiquement à jour.

Spécifications techniques :

Une bière est un objet qui possède les attributs suivants :

- Identifiant : obligatoire, valeur décimale entière, unique. Il s'agit d'un nombre qui représente la clé primaire. Il s'agit d'une valeur positive qui débute à 1 et qui est incrémentée à chaque nouvelle entrée dans la liste de bière. Ce champ est technique et non modifiable par l'utilisateur.
- Nom : obligatoire, chaîne de caractères. Il s'agit du nom commercial de la bière.
Exemple : L'angélus

- Commentaire : non obligatoire, chaîne de caractères. Ce champ permet d'ajouter un descriptif libre sur la bière. Exemple : bière ambrée au caractère malté, le plus souvent possédant des notes caramélisées, de grillé ou encore de noisette.
- Prix hors taxe (HT) : obligatoire, valeur décimale, précision : 2 chiffres après la virgule. C'est le prix de vente sans la taxe sur la valeur ajoutée. Exemple : 5.00 , l'unité est l'euro.
- Prix toute taxe comprise (TTC) : calculée, valeur décimale, précision : 2 chiffres après la virgule. C'est le prix de vente avec la taxe sur la valeur ajoutée. Exemple : 5.00, l'unité est l'euro. Ce champ n'est pas éditable puisqu'il est automatiquement déduit du prix hors taxe. Plus de précisions sur ce champ dans la suite du sujet.
- Degré d'alcool : obligatoire, valeur décimale, valeur positive comprise entre 0 et 70 Exemple : 6.5
- Type : obligatoire, chaîne de caractères, champ non libre. Il existe 4 types : DARK, BLONDE, IPA, BROWN, aucune autre valeur que celles énumérées ici n'est acceptée.
- Propriétaire : obligatoire, chaîne de caractères. Identification du propriétaire par une chaîne de caractères, le propriétaire est l'utilisateur qui a saisi la bière dans le formulaire.

Voici le même objet au format JSON :

```
{
  id: 2,
  name: "Filou",
  comment: "Très bonne bière",
  priceHT : 10,
  priceTTC: 12,
  alcoholDegree: 6.5,
  type: "BLONDE",
  owner: "defaultOwner"
}
```

La liste des bières contient simplement des objets « bière » dans un tableau.

Back-end :

Lien du git : <https://gitlab.com/noconsulting/noconsulting/stage/poc-shopping-back>

Un swagger est également disponible dans le projet.

Les développeurs sont disponibles pour vous présenter le projet.

Serveur de websocket :

Lien du git : <https://gitlab.com/Alexandreble/beersockets/-/tree/main>

Le développeur est disponible pour vous présenter le projet.