

Web Avancé: gestion de stocks

4A STI

EA Commerce Électronique

Groupe:

BOUCHAMMACH Soufian

DE GAILLANDE Clément

EL HAMDOUCHI Jade

LOAIZA ARBELAEZ Jhon Anderson

Préambule:

Le projet web-avancé que nous avons développé comporte une base simple pour la production d'une site web e-commerce, où des utilisateurs peuvent créer des comptes et commander des articles ajoutés à l'application par un administrateur.

Pour le bon fonctionnement de notre code, veuillez suivre la [guide d'utilisation \(III\)](#)

Notre projet fonctionne chez des membres ayant comme version:

- NPM 4.6.1 et 6.14

Sommaire:

I) Modèle de données

II) Liste des fonctionnalités

II.1) Comptes client et authentification

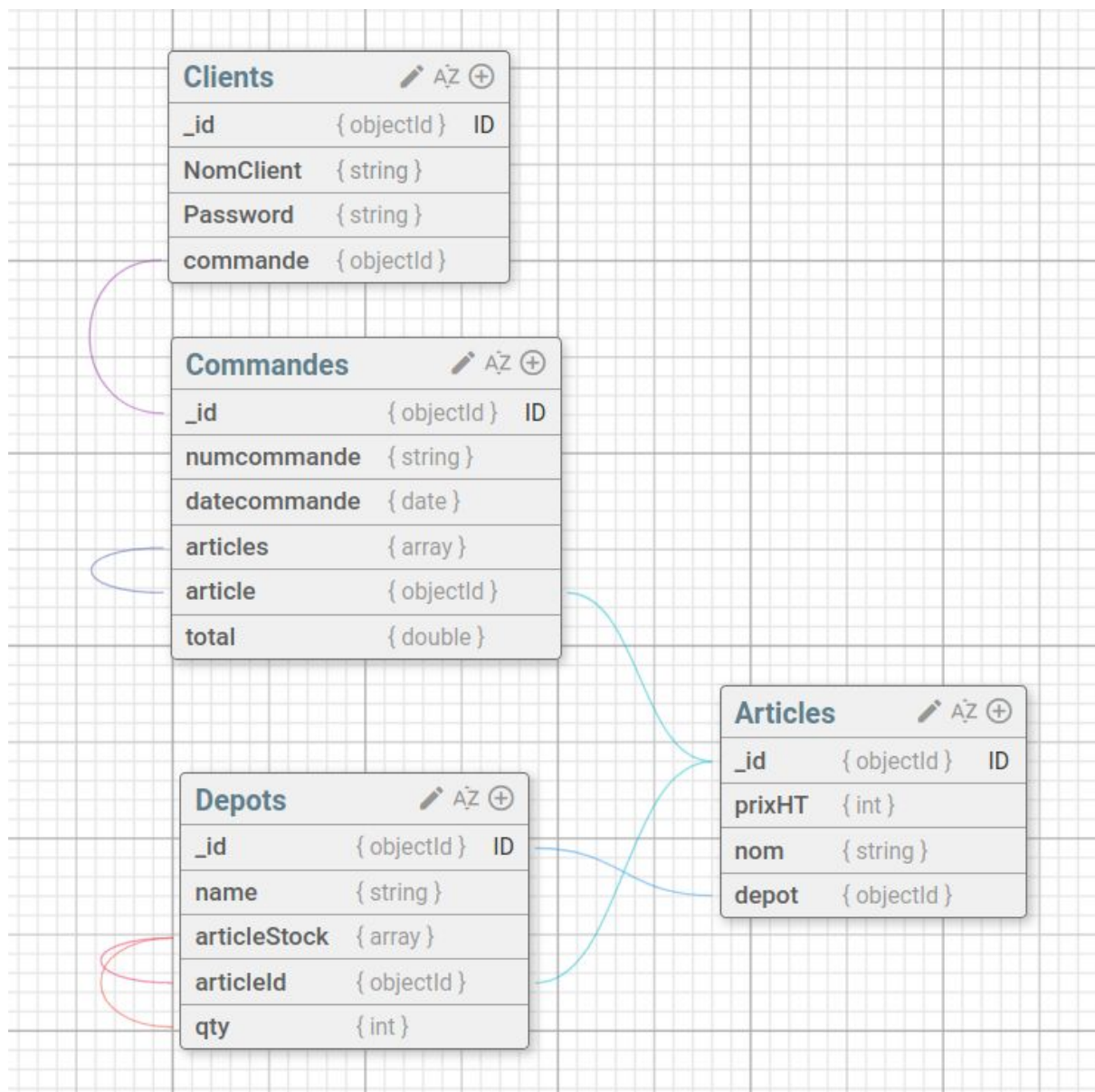
II.2) Compte administrateur / administration

III) Guide d'utilisation

IV) Points d'amélioration

V) Contributions? Références?

I) Modèle de données



II) Fonctionnalités développées

II.1) Comptes client et authentification

Les comptes clients correspondent à la collection clients en BDD.

Deux scénarios pour l'authentification:

- 1) L'authentification initiale se fait lorsque le client entre son login et mot de passe sur le front-end ReactJS. Une requête POST est alors envoyée vers une route /login du back-end qui compare ce login et mot de passe à ceux existant en BDD. Si correspondance: un jwt authentifiant le client est généré et envoyé au client.

- 2) Au moment d'effectuer une action (passer commande, ...), le jwt est envoyé depuis le front-end vers le back-end qui vérifie sa validité avant de prendre en compte l'action..

Lors de la création des comptes clients les mot de passes sont enregistrés hashés.

Nous nous sommes aidés du package NPM [jsonwebtoken](#) pour la génération/signature et vérification des tokens.

Du coté de l'interface client, elle se présente ainsi :

User name: yayo

Log-out!

CATALOGUE

fruits

Nom: pomme
Prix: 3 €
Ajouter au panier: 0 (984 disponibles)
+ -

Nom: banana
Prix: 10 €
Ajouter au panier: 0 (952 disponibles)
+ -

Nom: banano
Prix: 2 €
Ajouter au panier: 0 (68 disponibles)
+ -

Nom: cebolla
Prix: 4 €
Ajouter au panier: 0 (988 disponibles)
+ -

Nom: coconut
Prix: 5 €
Ajouter au panier: 0 (975 disponibles)
+ -

Nom: pera
Prix: 3 €
Ajouter au panier: 0 (Aucun disponible)
+ -

Nom: papaya

Nom: mango

Panier

Aucun article sélectionné

Commander!

Total: 0 €

show my commands

N°	Command Number	Articles	Command Date	Total
----	----------------	----------	--------------	-------

On peut sélectionner les différents articles regroupés dans les différents dépôts pour les mettre dans notre panier :

Log-out!

CATALOGUE

fruits

Nom: fraise
Prix: 3 €
Ajouter au panier: 3
(47 disponibles)

Nom: kiwi
Prix: 6 €
Ajouter au panier: 0
(80 disponibles)

Nom: mango
Prix: 3 €
Ajouter au panier: 2
(13 disponibles)

vetements

Nom: jupe
Prix: 16 €
Ajouter au panier: 0
(20 disponibles)

Nom: chapeau
Prix: 12 €
Ajouter au panier: 2
(18 disponibles)

Panier

Product: fraise x3 = 9 €

Product: chapeau x2 = 24 €

Product: mango x2 = 6 €

Product: raquette x3 = 72 €

Commander!

Total: 111 €

show my commands

N°	Command Number	Articles	Command Date	Total
----	----------------	----------	--------------	-------

Puis on peut commander et afficher un bon de commande :

Log-out!

CATALOGUE

fruits

Nom: fraise
Prix: 3 €
Ajouter au panier: 0
(47 disponibles)

Nom: kiwi
Prix: 6 €
Ajouter au panier: 0
(80 disponibles)

Nom: mango
Prix: 3 €
Ajouter au panier: 0
(13 disponibles)

vetements

Nom: jupe
Prix: 16 €
Ajouter au panier: 0
(20 disponibles)

Nom: chapeau
Prix: 12 €
Ajouter au panier: 0
(18 disponibles)

Panier

Aucun article sélectionné

Commander!

Total: 0 €

show my commands

N°	Command Number	Articles	Command Date	Total
1	5e90bd87df24d366dd0d34b5	<ul style="list-style-type: none"> nom: fraise quantity: 3 prix: 3 € nom: mango quantity: 2 prix: 3 € nom: chapeau quantity: 2 prix: 12 € nom: raquette quantity: 3 prix: 24 € 	4/10/2020	111 €

II.2) Compte administrateur / administration

Le compte administrateur doit absolument être créé avec le nom d'utilisateur "admin".

L'administrateur a accès à une interface administrateur :

Welcome to CSJJ app!

User name: admin

Log-out!

Ajouter un nouvel article

Depot: fruits

Nom de l'article:

Prix: 0

Quantité 0

Ajouter un article

Ajouter un dépôt

Nom du dépôt:

Ajouter un dépôt

Il peut notamment ajouter un article :

Ajouter un nouvel article

Depot: test

Nom de l'article:

Prix: 0

Quantité 0

Ajouter un article

limites: Si l'administrateur mets des strings dans le champs prix ou quantité, le serveur deviendra en panne. Il faut bien utiliser des nombres. Les articles ne sont pas modifiables par l'admin: une fois que l'admin enregistre un article, il ne pourra pas changer son prix, quantité ou nom; ces changes doit être faites directement dans la base de données.

L'ajout d'article se fait dans un dépôt. Il peut aussi ajouter un dépôt :

Ajouter un dépôt

Nom du dépôt:

Ajouter un dépôt

III) Guide d'utilisation

Ce projet utilise les techno suivantes : nodejs,react et mongoDB. Nous recommandons installer mongoDB compass pour afficher et examiner la base de données d'une manière plus facile.

Pour lancer l'application, lancer la commande "**npm install**" depuis les dossiers **frontend** et **backend** dans un terminal. Puis **npm run dev** depuis le dossier backend et **npm start** depuis le dossier frontend.

À ce moment, notre base de données est vide. On n'a pas des utilisateurs ni d'articles. Alors il faut tout d'abord créer notre administrateur remplira notre application avec de dépôts et ses articles.

1. Allez dans le lien **sign up** et créez un utilisateur avec le nom **admin** et un mot de passe dont vous pouvez facilement vous rappeler.

2. Allez dans la partie **Ajouter un dépôt**. Mettez le nom que vous voulez.
3. Ajoutez des articles dans la partie **Ajouter un nouvel article**. Rappelez-vous des limites de cette application en bien mettant des nombres entiers dans les labels 'prix' et 'quantité'.
4. Faites un log-out ou ouvrez un nouvel onglet avec l'adress localhost:3000, puis créez un nouveau utilisateur dans sign-up. logez up avec cet utilisateur.
5. Vous pouvez désormais faire des commandes!

Welcome to CSJJ app!

Sign up or login to make orders.

User name:

- [Login](#)
- [Sign up](#)

Username:

Password:

Repeat password:

Puis se loguer avec et mettre à jour les stocks.

D'autres utilisateurs pourront alors créer leurs comptes et passer des commandes.

IV) Points d'amélioration

- Si deux utilisateurs qui soient en train de faire des achats, presque en même temps, commandent un article dont il y reste qu'une seule quantité, il y aura une problème de cohérence dans la base de données car notre application a un limite dans ce point spécifique. Les commandes seront bien faites du côté client, mais en réalité l'application ne compte plus avec la quantité nécessaire pour servir le client qui ait commandé quelques seconds après.
- le CSS et design qu'on n'a pas développé ni repris d'un modèle existant, la livraison des fonctionnalités ayant pris du retard
- la partie administration se repose sur le nom d'utilisateur

- manque de visibilité des articles présents dans la base de données du côté administrateur
- un import depuis un fichier excel, csv, etc... pour les stocks par l'administrateur plutôt que manuel

V) References

Authentication:

- [Introduction to JSON Web Tokens](#)
- [JWTs in React for Secure Authentication](#)
- [Authentication for your React and Express application](#)

Formulaires & user input:

- [Forms](#)
- [Building a "Mask Toggle" Password Input Component w/ React and Material UI](#)
-

Back-end:

- [Create a simple and stylish Node Express app](#)
- [Mongoose: Get full list of users](#)
- [Password Authentication with Mongoose Part 1](#)
- <https://devdocs.io/javascript/>
-