

TP 3 – SPA - ECOMMERCE

PRÉSENTÉ À MARCOS SANCHES



RÉALISÉ PAR
JULIE RENAUD



LIEN GITHUB

<https://github.com/JulieDeRen/TP3-SPA>



1. DESCRIPTION DU PROJET

Voici une application web d'une page conçue avec le cadre Vue.js du côté client et Node.js du côté serveur. Les données sont lus, affichées, modifiées et effacées directement sur la base de donnée.

2. FONCTIONNALITÉS

Les fonctionnalités sont les suivantes :

- Affichage des données sur la page d'accueil
- Lorsqu'on clique sur une carte, une fiche détaillé s'affiche
- On peut ajouter des éléments soit du catalogue de la page d'accueil ou de la fiche vers le panier
- Le panier est mis à jour automatiquement
- On peut retirer des éléments du panier
- On peut modifier un item en cliquant sur le bouton modifier qui se trouve sur la fiche de chaque item
- On peut ensuite effacer l'item également
- On peut ajouter un nouvel item qui sera enregistré dans la base de données et affiché automatiquement.

3. LIGNES DE COMMANDES

Les lignes de commandes pour installer le site sont les suivantes :

- Vue CLI nécessite Node.js version >8.9 / installer à partir du lien <https://nodejs.org/en/>
- Voir la version de node.js : # node -v
- Installer le nouveau paquet Vue.js # npm install -g @vue/cli
- Voir la version # vue --version
- Créer un nouveau projet : # vue create tp3-spa-ecommerce-vue
Nom du projet en minuscule
Installation manuelle
Installer Babel, routeur et linter/formatter
- Aller sur le repertoire du projet créé : # cd tp3-spa-ecommerce-vue
- Ajouter un dossier côté client : # mkdir projet-vue
- Installer bootstrap du côté client : npm install bootstrap jquery popper.js
- Installer Axios pour les requêtes au serveur : # npm install axios
- Ajouter un serveur au projet : # mkdir server
- Aller sur le serveur : # cd server
- Initialiser le nouveau serveur Node.js et créer le package.json
- Compléter la configuration de l'initialisation du serveur et package.json avec les infos suivantes :
 - package name: (server) server
 - version: (1.0.0)
 - description: server with nodejs, express, sequelize, cors, mysql
 - entry point: (index.js) server.js
 - test command: git repository: keywords: node, express, sequelize, mysql, cors
 - author:
 - license: (ISC)
- Enregistrer le package.json
- Installer les modules nécessaires au serveur : # npm install express sequelize mysql2 body-parser cors --save
- Configurer les fichiers du serveur dans l'éditeur de texte
- Tester le serveur : # node server.js

- Démarrer le serveur local côté serveur : `# npm start server`
- Démarrer le serveur local côté client : `# npm run serve`

4. RÉFÉRENCES

Toutes les références utilisées sont inscrites dans le fichier Readme.md dans la racine du projet et anotées dans le code.