

Supinfo

JEANSSTORE

DOCUMENTATION TECHNIQUE



JEANSSTORE

LES MEILLEURS JEANS
POUR VOS ENFANTS

07-01-2022

Table des matières

I.	Arborescence des fichiers.....	2
II.	Description du fonctionnement global	5
III.	Prérequis.....	7
IV.	Installation	7

DOCUMENTATION TECHNIQUE

I. Arborescence des fichiers

Le repertoire de JEANSTORE

- Assets (pour la gestion des fichiers css et js grâce au module webpack encore de symfony). Dans ce projet nous avons importer les fichiers css à la main
 - Controllers
 - Styles
- Bin
- Config
 - Packages : contient les fichiers de configurations des librairies utilisés
 - Routes : Indique à symfony les chemins pour la resolution des routes
présentent dans le dossier controller
- Migrations
 - Les requêtes pour la base de données pour la mise à jour de la base de données
- Public
 - Build : contient les fichiers de la librairie webpack encore
 - Deco : contient les fichiers css, js utilisés
 - Uploads : contient les images des produits
 - Index.php : contient le mapping du site
- Src
 - Controller : contient les fichiers qui gère le backend du site, les différents url et les fonctions utiles pour ces url.

- Entity : contient les class liées aux entités de la base de données
 - EventSubscriber : contient la class pour gérer le bouton de changement de langue du site
 - Extensions : contient la class pour gérer la requête fulltext search sur le site
 - Form : contient les prototypes des formulaires utilisées pour la création, édition et mise à jour de la base de données
 - Repository : contient les class pour gérer les requêtes « GET, POST, UPDATE, DELETE » des entités de la base de données
 - Security : contient les fichiers pour l'authentification des utilisateurs
- Templates (contient les vues)
- Cart : contient les vues relatives au panier
 - Cart.html.twig
 - Checkout.html.twig
 - Categories : contient vues pour la création d'une catégorie pour les produits
 - Edit.html.twig
 - Index.html.twig
 - New.html.twig
 - Show.html.twig
 - _form.html.twig (contient le formulaire lié à l'entité)
 - _delete_form.html.twig (contient le formulaire pour la suppression lié à l'entité)
 - Coupons : les vues pour la création des coupons de réduction
 - Edit.html.twig
 - Index.html.twig
 - New.html.twig
 - Show.html.twig
 - _form.html.twig (contient le formulaire lié à l'entité)

- `_delete_form.html.twig` (contient le formulaire pour la suppression lié à l'entité)
- Image : contient les vues pour l'édition des images
 - `Edit.html.twig`
 - `Index.html.twig`
 - `New.html.twig`
 - `Show.html.twig`
 - `_form.html.twig` (contient le formulaire lié à l'entité)
 - `_delete_form.html.twig` (contient le formulaire pour la suppression lié à l'entité)
- Orders : contient les vues concernant les commandes effectués
 - `Index.html.twig`
 - `_form.html.twig` (contient le formulaire lié à l'entité)
 - `_delete_form.html.twig` (contient le formulaire pour la suppression lié à l'entité)
 - `Show.html.twig`
- Products : contient les vues pour la gestion des produits
 - `Edit.html.twig`
 - `Index.html.twig`
 - `New.html.twig`
 - `Show.html.twig`
 - `_form.html.twig` (contient le formulaire lié à l'entité)
 - `_delete_form.html.twig` (contient le formulaire pour la suppression lié à l'entité)
- Registration : contient la vue pour le formulaire d'inscription
 - `Register.html.twig`
- Security : contient la vue pour le formulaire de connexion
 - `Login.html.twig`


- Store : contient les pages de la boutique
 - Admin.html.twig
 - Contact.html.twig
 - Index.html.twig
 - Shop.html.twig
- Users : contient les vues pour les informations et l'édition d'un utilisateur du site
 - Edit.html.twig
 - Index.html.twig
 - New.html.twig
 - Show.html.twig
 - __form.html.twig (contient le formulaire lié à l'entité)
 - __delete__form.html.twig (contient le formulaire pour la suppression lié à l'entité)
- Base.html.twig (le template de base qui contient les fichiers js, css et la navigation)
- Tests (Prévu pour les tests unitaires)
- Translations (Prévu pour les fichiers de traduction de la langue du site)
- Var (contient les log et le cache)
- Vendor (contient les dépendances du site)
- .env (contient les configurations pour la connexion à la base de données)

II. Description du fonctionnement global


🚦 Le site pointe sur la page d'accueil shop.html.twig lorsqu'il est lancé; le fichier base.html.twig est inclus dans les pages du site afin de faciliter la navigation et l'utilisation des fichiers css et js.

Sur la page d'accueil on a le menu de navigation qui contient les rubriques :

- Accueil : dirige sur la page de sélection des produits
- Commandes : les produits commandés par un utilisateur X
- Contact : les informations pour contacter l'entreprise
- Compte :
 - Connexion : Pour se connecter ou créer un compte utilisateur
- Panier qui contient les produits sélectionnés
 - Le panier est automatiquement vidé dans les cas suivants :
 - L'utilisateur a payé les produits
 - 24 h se sont écoulés depuis la dernière heure d'ajout d'un produit.
 - Tout ajout d'un article dans le panier redirige sur la page panier, on a la possibilité de supprimer le panier ou d'augmenter la quantité des produits à acheter; il est possible d'obtenir une réduction grâce au coupon que l'administrateur génère.
 - Le bouton « payer » redirige vers la page « checkout.html.twig » où on a un dernier aperçu des articles que l'utilisateur s'apprête à payer si et seulement si l'utilisateur est connecté sinon l'utilisateur est rediriger vers la page de connexion où il doit obligatoirement se connecter ou s'inscrire s'il n'a pas de compte et fournir ses adresses de livraisons et de facturations.

 Sur la page d'accueil on a une zone de recherche des produits par catégorie et par nom.

L'inscription d'un nouvel utilisateur le redirige vers la page d'accueil qui est celle de la boutique.

-  Après connexion, si l'utilisateur connecté à un rôle ADMIN_ROLE, il peut accéder à la page admin visible dans le drop down de compte.
- Il peut voir tous les utilisateurs inscrits du site
 - Créer des produits pour la boutique

- Voir toutes les commandes et générer les reçus en format Pdf pour chaque commande.
- Voir et créer des coupons pour des réductions.

III. Prérequis

Pour fonctionner le serveur à besoins de ses éléments suivants :

Système d'exploitation	Windows, linux ou Mac
Moteur apache	Apache/2.4.51
PHP	Supérieur ou égale à 8.0.1
Mysql	10.4.21-MariaDB
Symfony	Supérieur ou égale à Symfony CLI version v4.26.11
Composer	Supérieur ou égale à version 2.1.14

IV. Installation

Pour Windows /Mac /Linux :

1. Installer Xampp : <https://www.apachefriends.org/fr/download.html>
2. Installer Composer <https://getcomposer.org/doc/00-intro.md> (Partie Installation Windows/ Mac /Linux)
3. Installer Symfony <https://symfony.com/download> (Windows /Mac /Linux)
4. Mettre le dossier JeanStore dans le dossier Htdocs de Xampp
5. Ouvrir localhost/phpMyAdmin
 - a. Créer une base de données mysql avec pour nom : store
 - b. Importer dans la base de données Mysql store précédemment créer la base de données joint au dossier JeanStore « store.sql ».
6. Ouvrir le dossier JeanStore dans le répertoire Htdocs de Xampp avec un éditeur de texte puis ouvrir le fichier « .env » à la racine du dossier.

- a. Fournir les informations de connexion à la base de données mysql (Nom d'utilisateur et mot de passe) à la ligne 32 et 33 du fichier « .env »
 - b. Enregistrer et fermer le fichier **.env**
7. Dans le terminal de l'éditeur ou dans le terminal (ex PowerShell) de la machine, on se positionne dans le dossier jeanStore (c :> cd jeanStore), et on tape les commandes suivantes:
- a. **Composer update** (pour installer toutes les librairies et les dépendances requises par symfony et le projet)
 - b. **Symfony server:start** (pour lancer le serveur Symfony après l'installation des dépendances)
 - c. **On ouvre le navigateur et on saisit l'url https://localhost:8000/ pour accéder au site JeanStore.**
 - d. **Identifiant du compte administrateur par défaut : symf@gmail.com
mot de passe : admin123456 pour se connecter en tant qu'administrateur.**